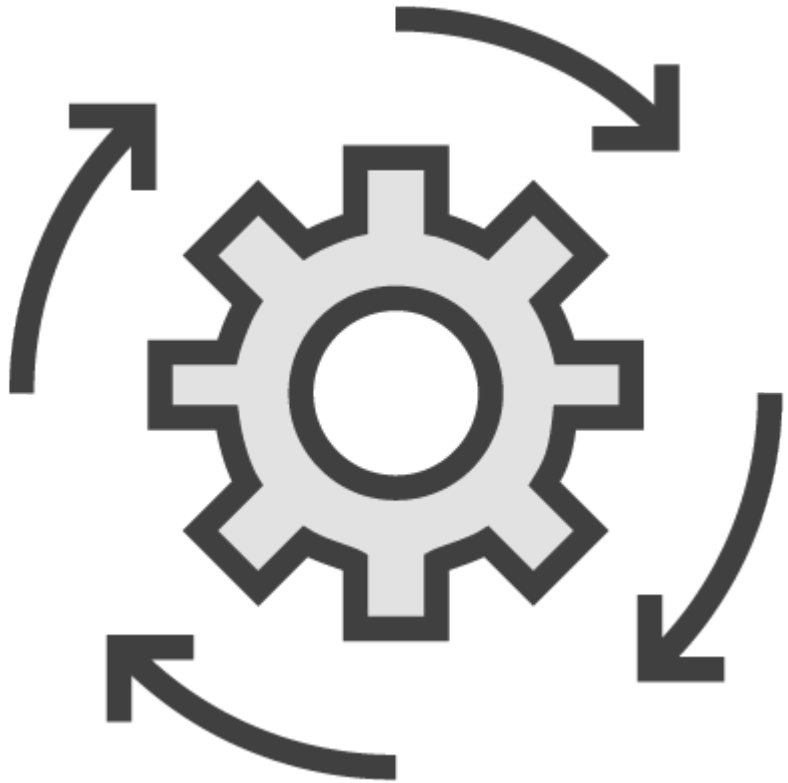
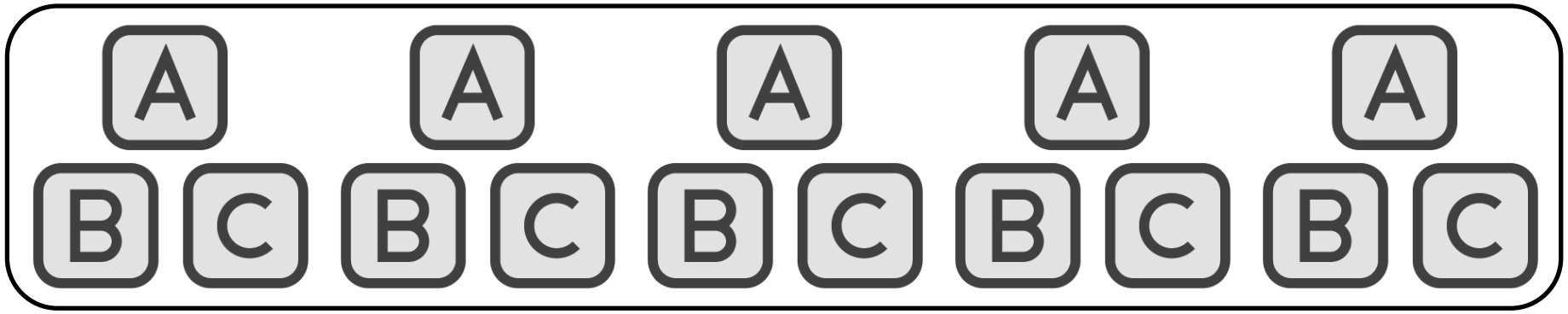
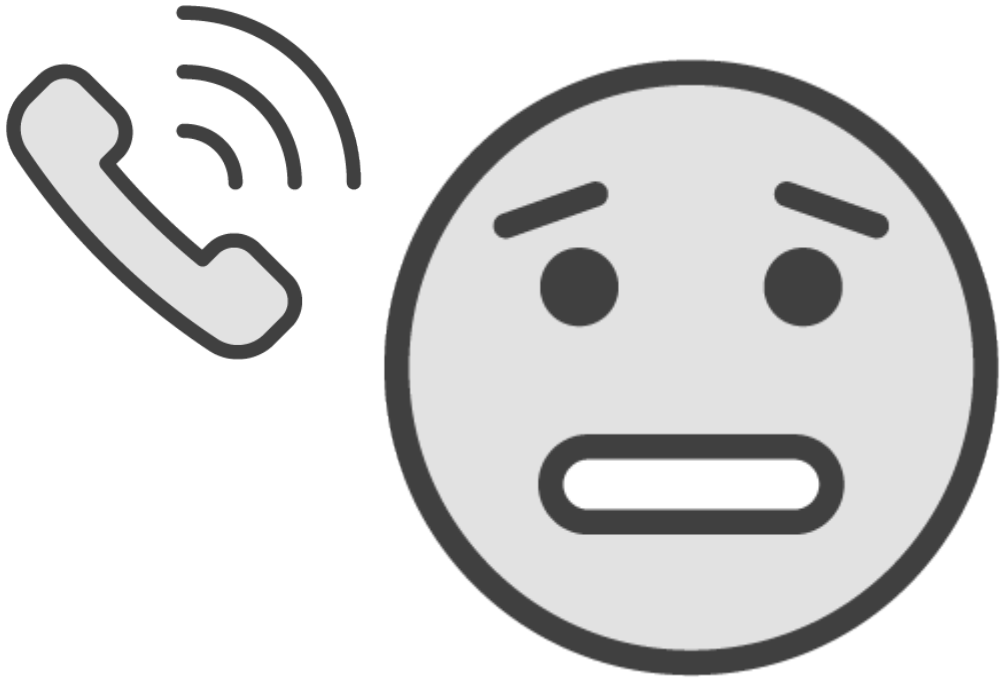
Beyond the Basics: Customization Through Configuration

Justin Pihony
DEVELOPER SUPPORT MANAGER @ LIGHTBEND

@JustinPihony




A A A A A
B C B C B C B C B C

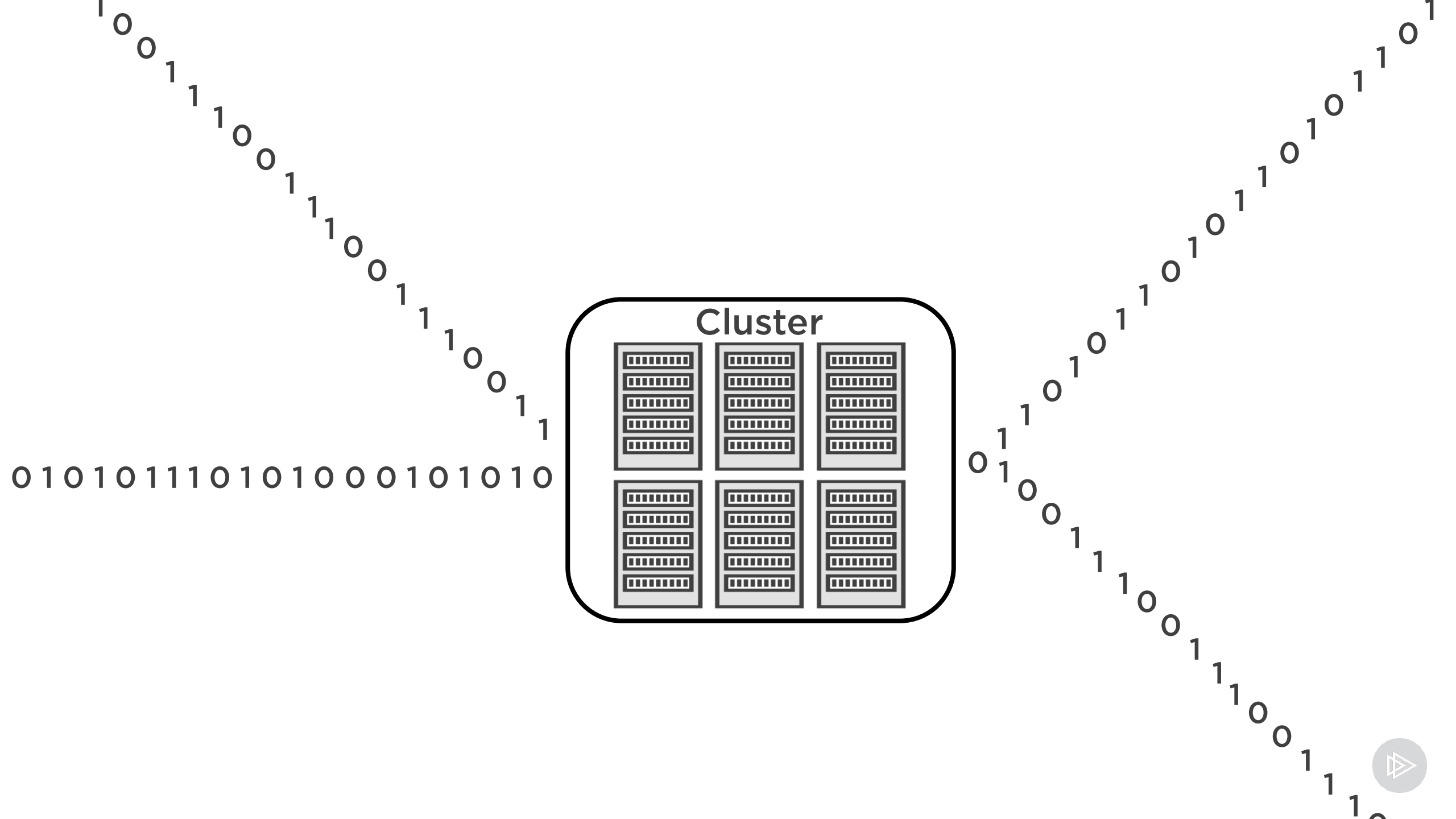


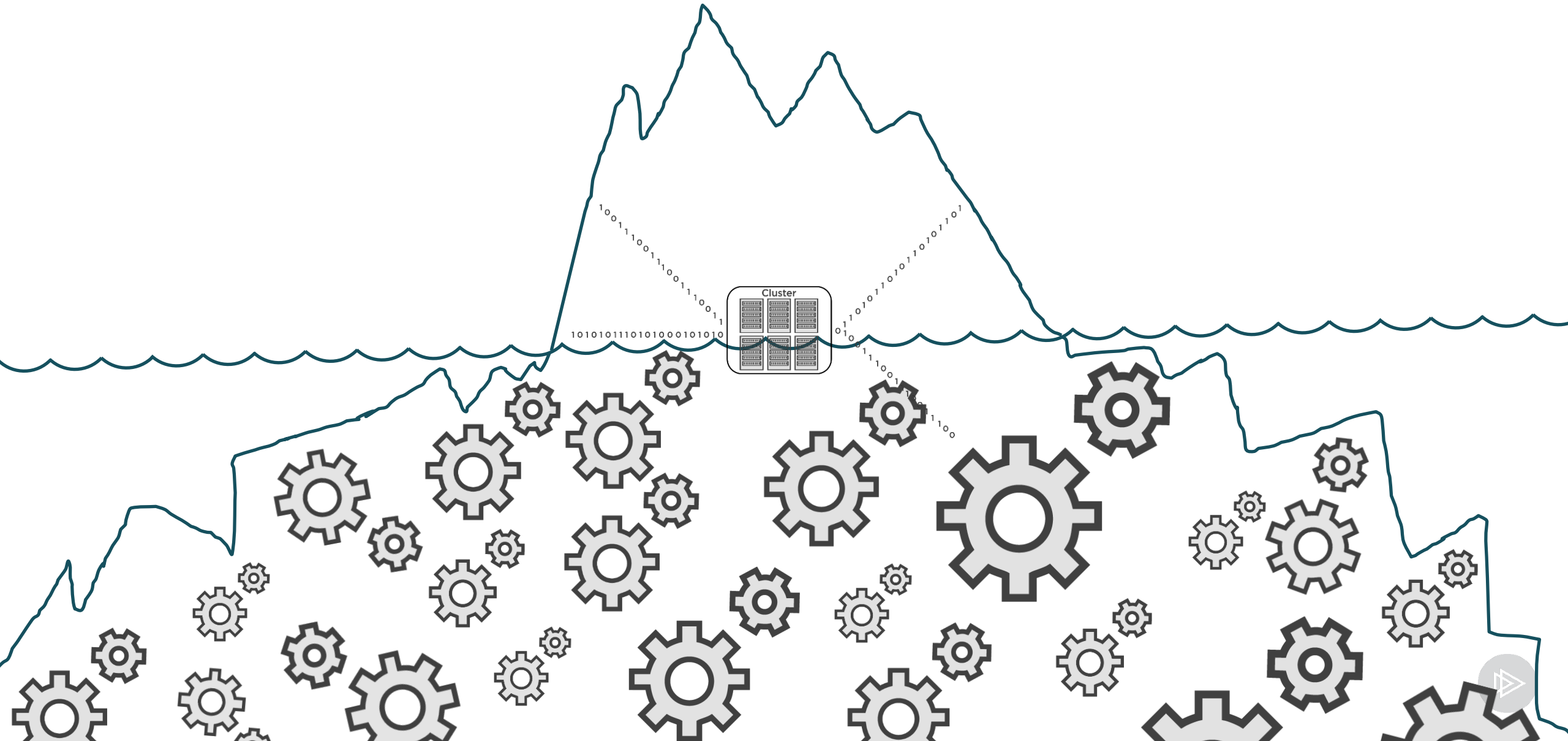
Configuration

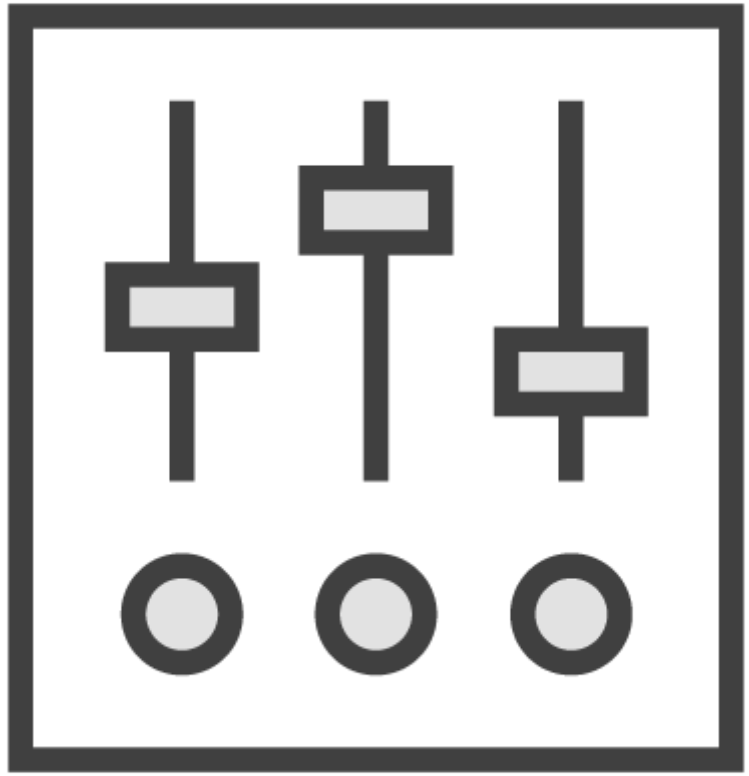
Broker

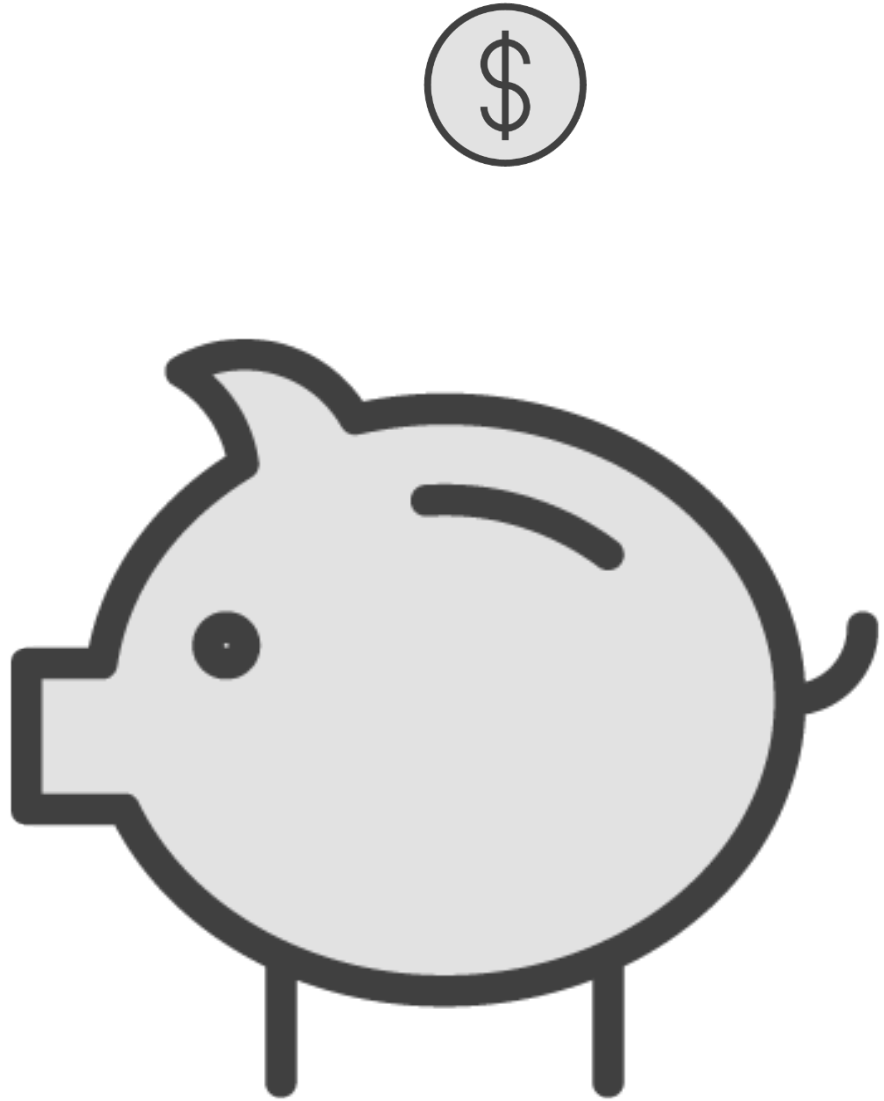
- Scale
- Parallelization
- Fault Tolerance
- Data Retention

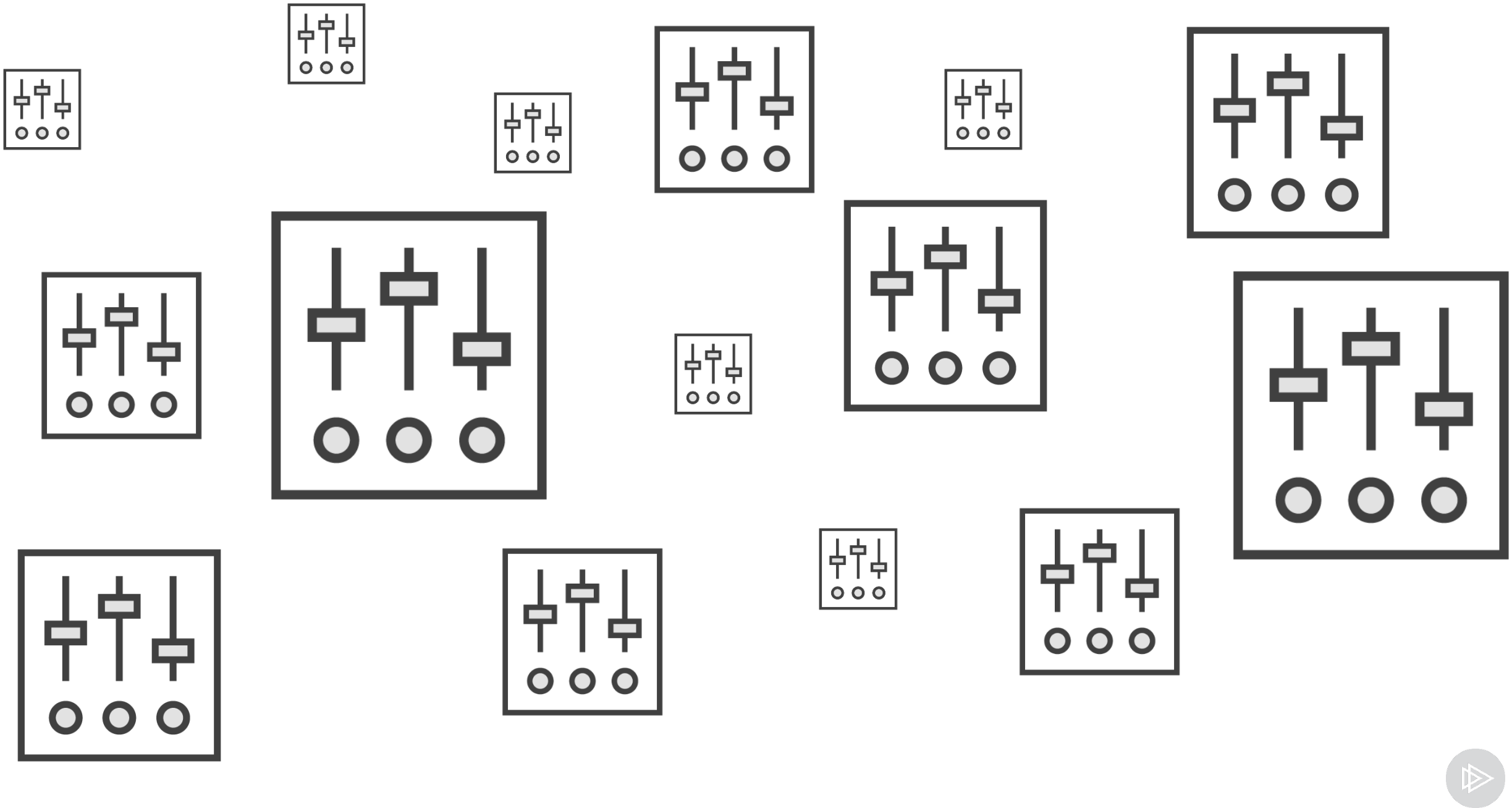
Detangling the Configuration Overload


Cluster


Cluster







Configuring the Broker: Core Configuration and Management

How?

kafka-server-start **config/server.properties**

1.1.0+

kafka-configs --bootstrap-server localhost:9092 ...

NAME	DESCRIPTION	TYPE	DEFAULT	VALID VALUES	IMPORTANCE	DYNAMIC UPDATE MODE

read-only

Requires server.properties change and broker restart

per-broker

kafka-configs must provide `entity-name [BROKER.ID]`

cluster-wide

kafka-configs *can* provide `entity-default`

Minimum Broker Configurations

- **broker.id**
- **zookeeper.connect**
- **log.dirs**

Important Broker Configurations

listeners

format → listener_name://host_name:port

default → PLAINTEXT://:9092

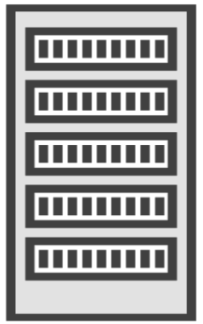
Important Broker Configurations

message.max.bytes >= **fetch.message.max.bytes**

default → 1000012 (1 mb)

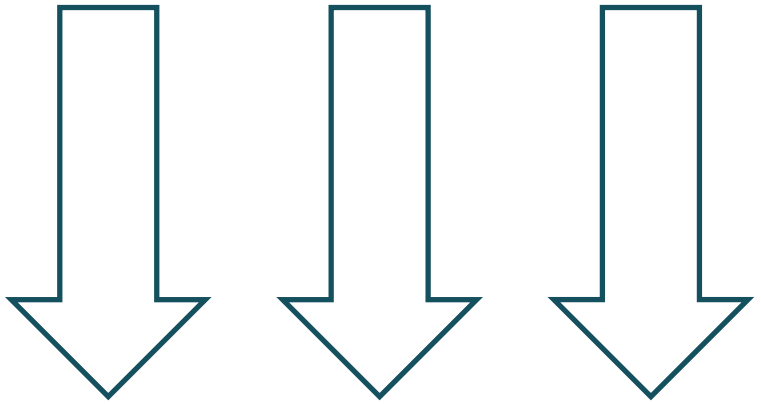
topic override = max.message.bytes

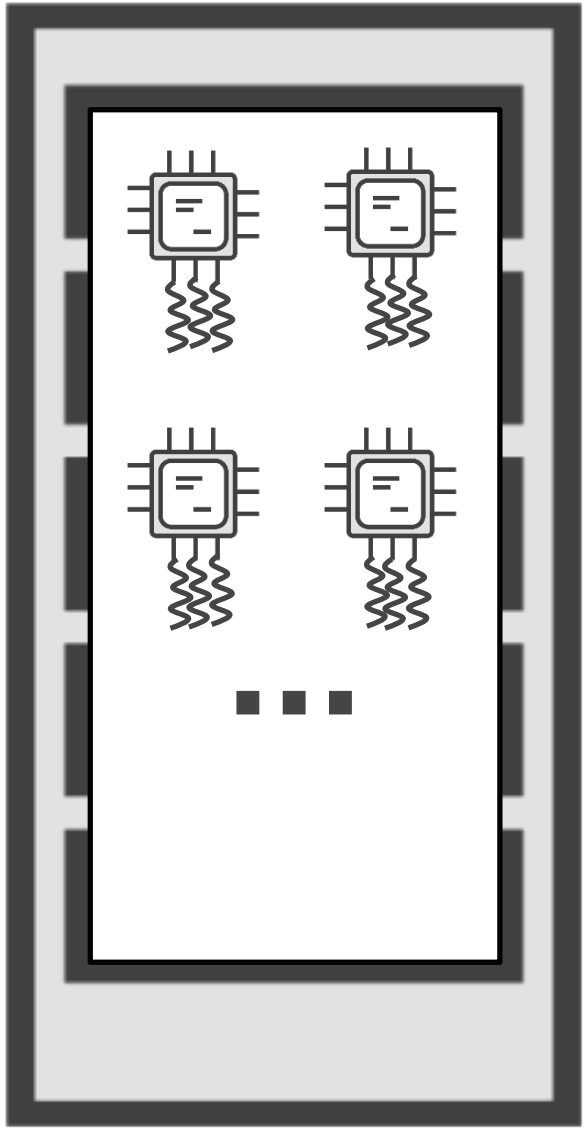
BROKER



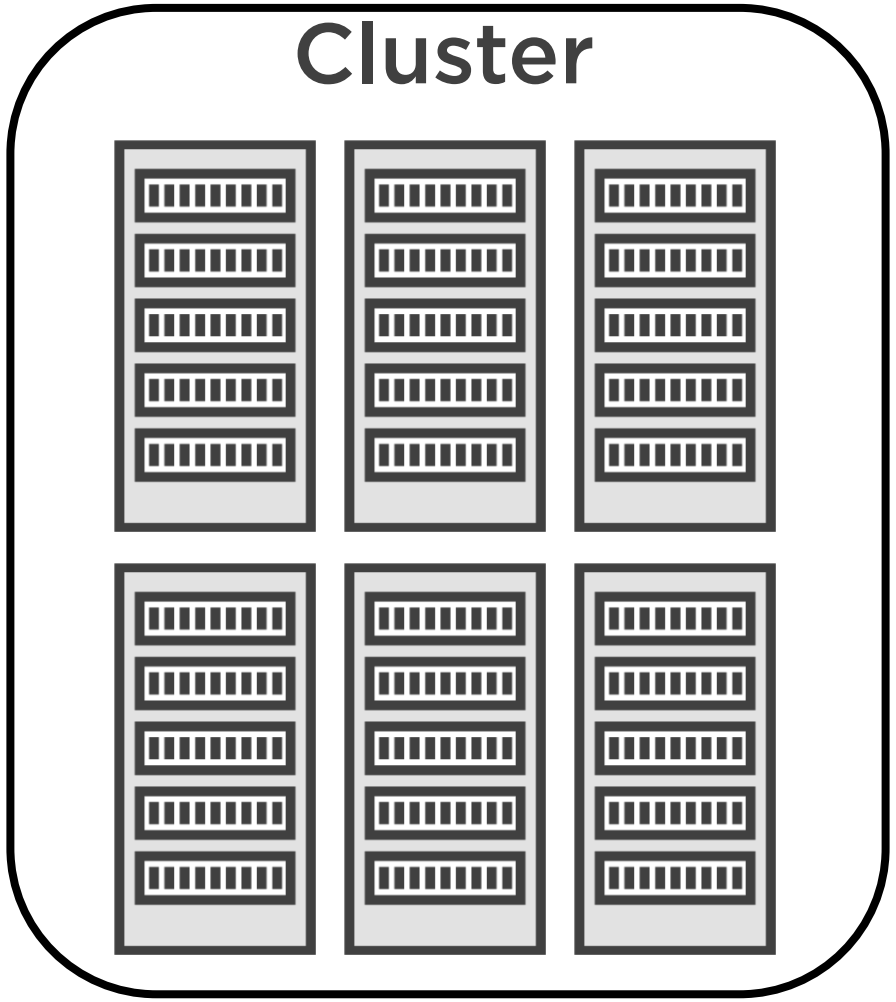
CONSUMER

Configuring the Broker: Parallelizing via Partitions

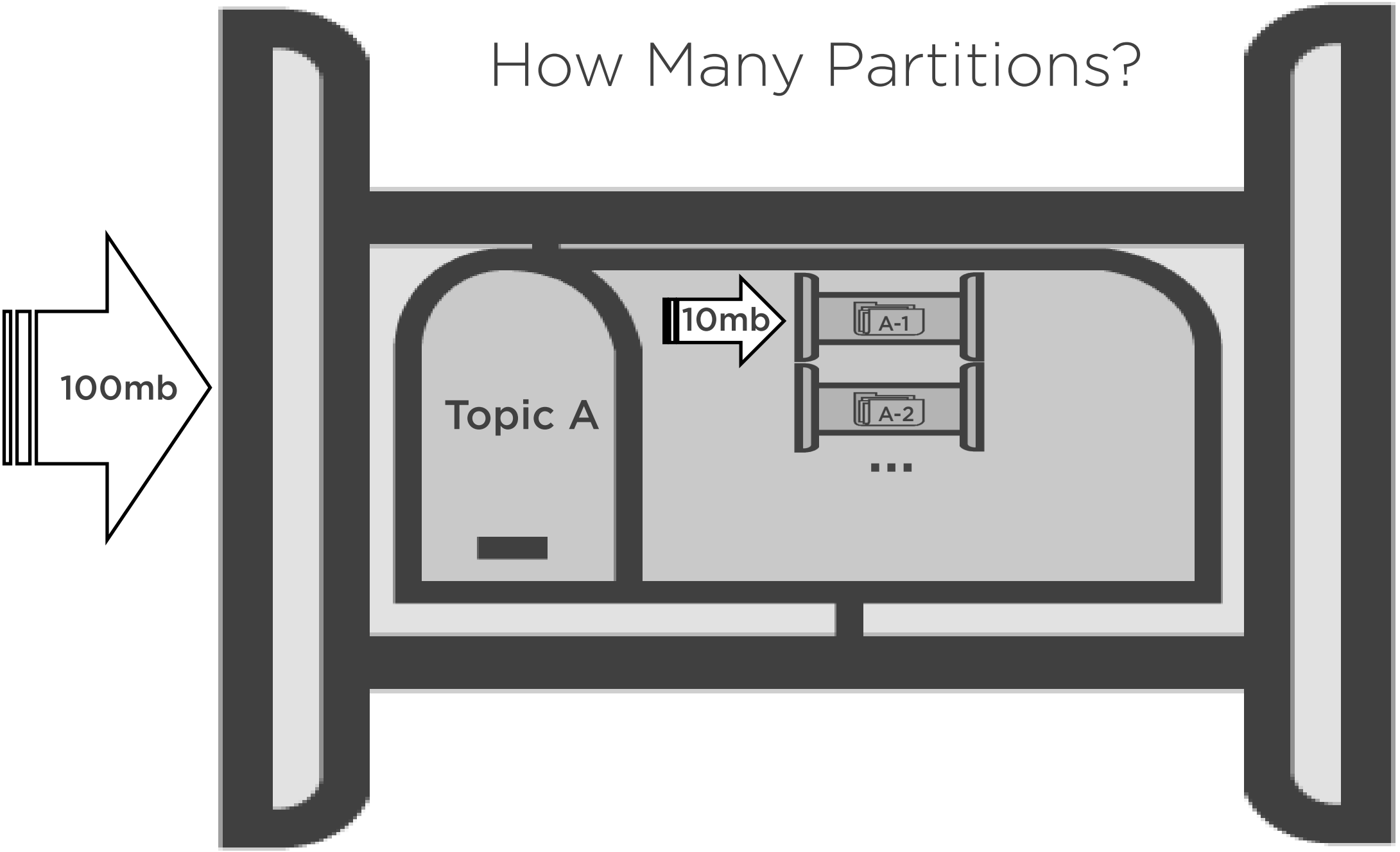




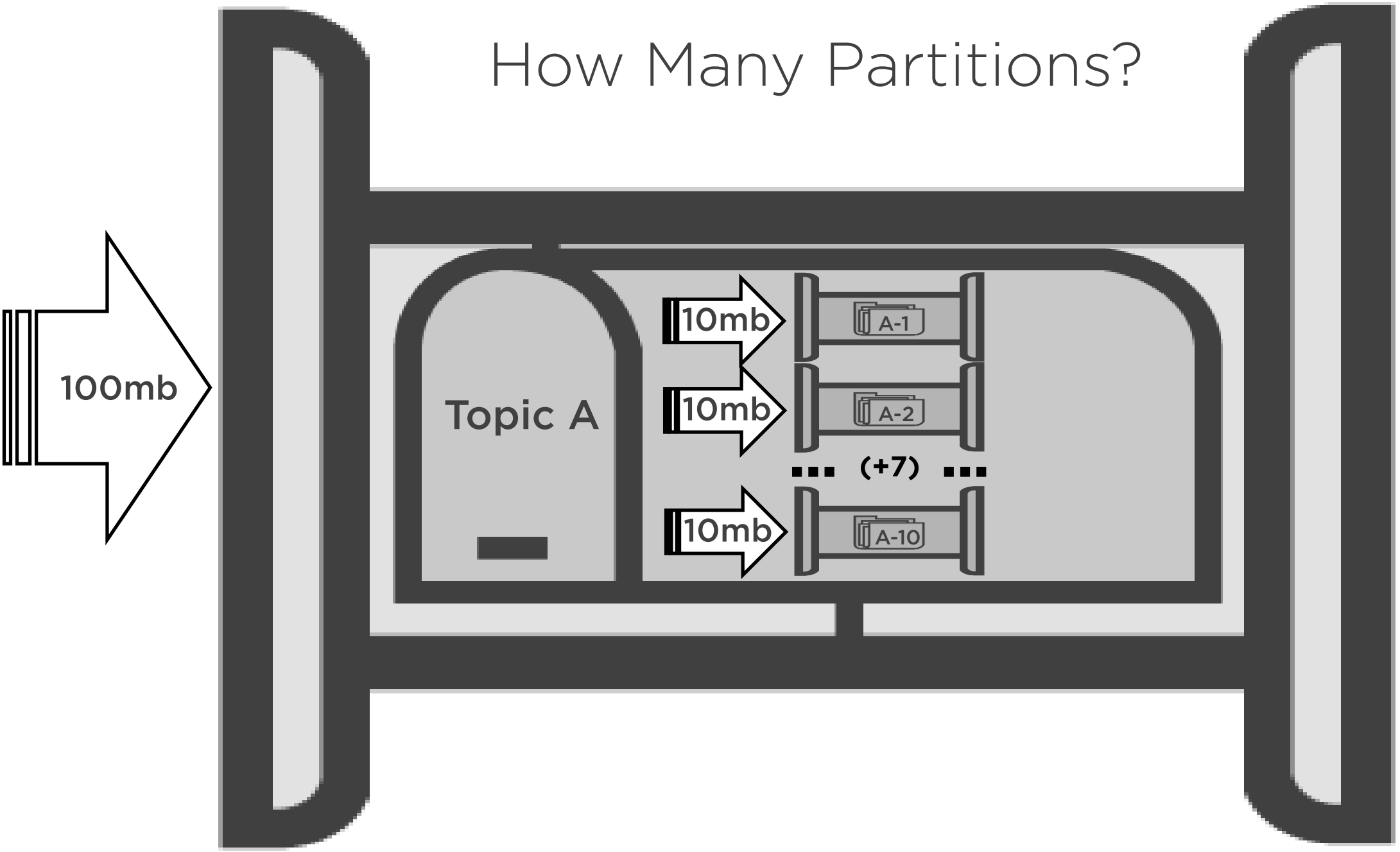
num.partitions = [MULTIPLE OF 6]

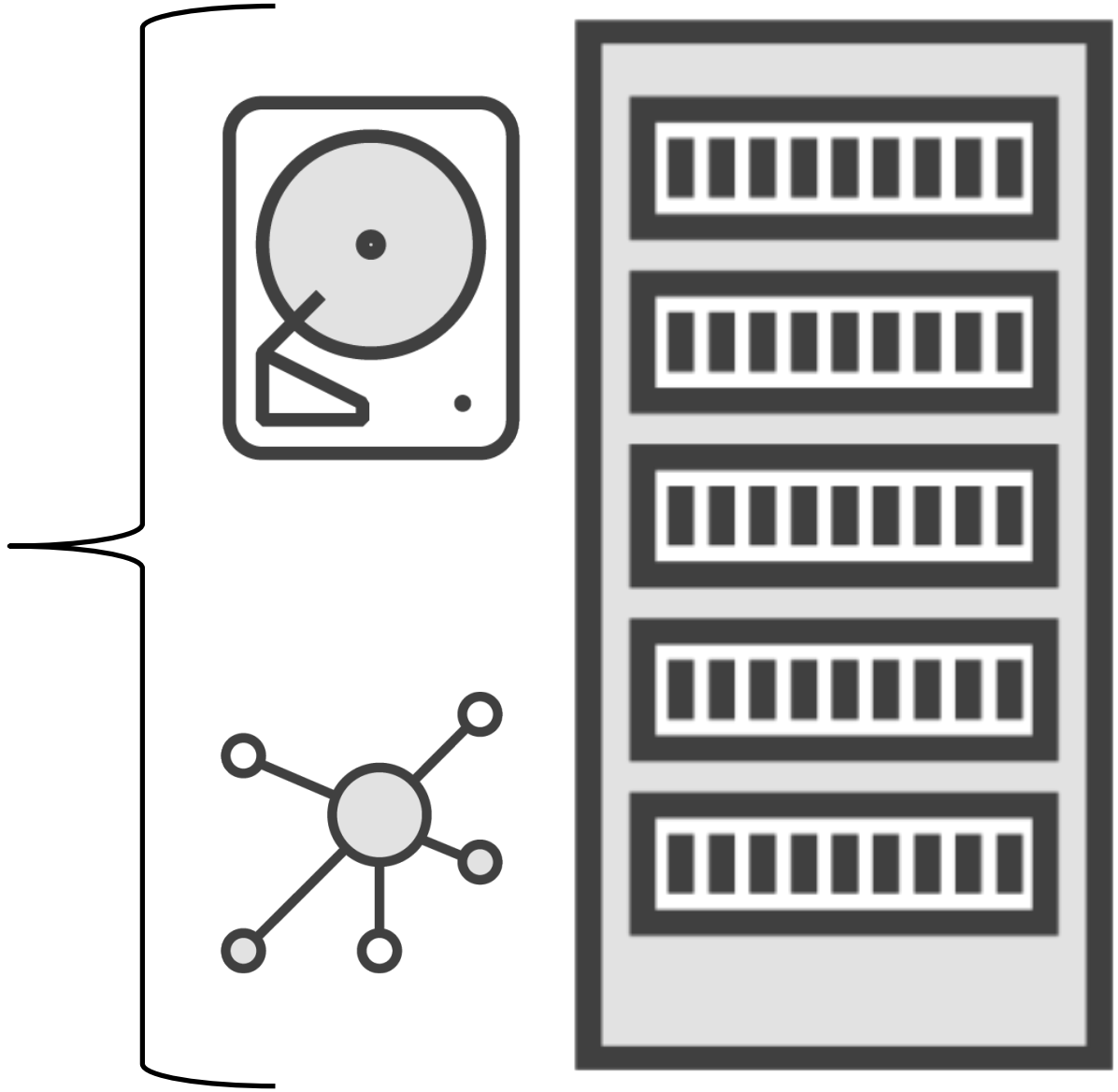


How Many Partitions?

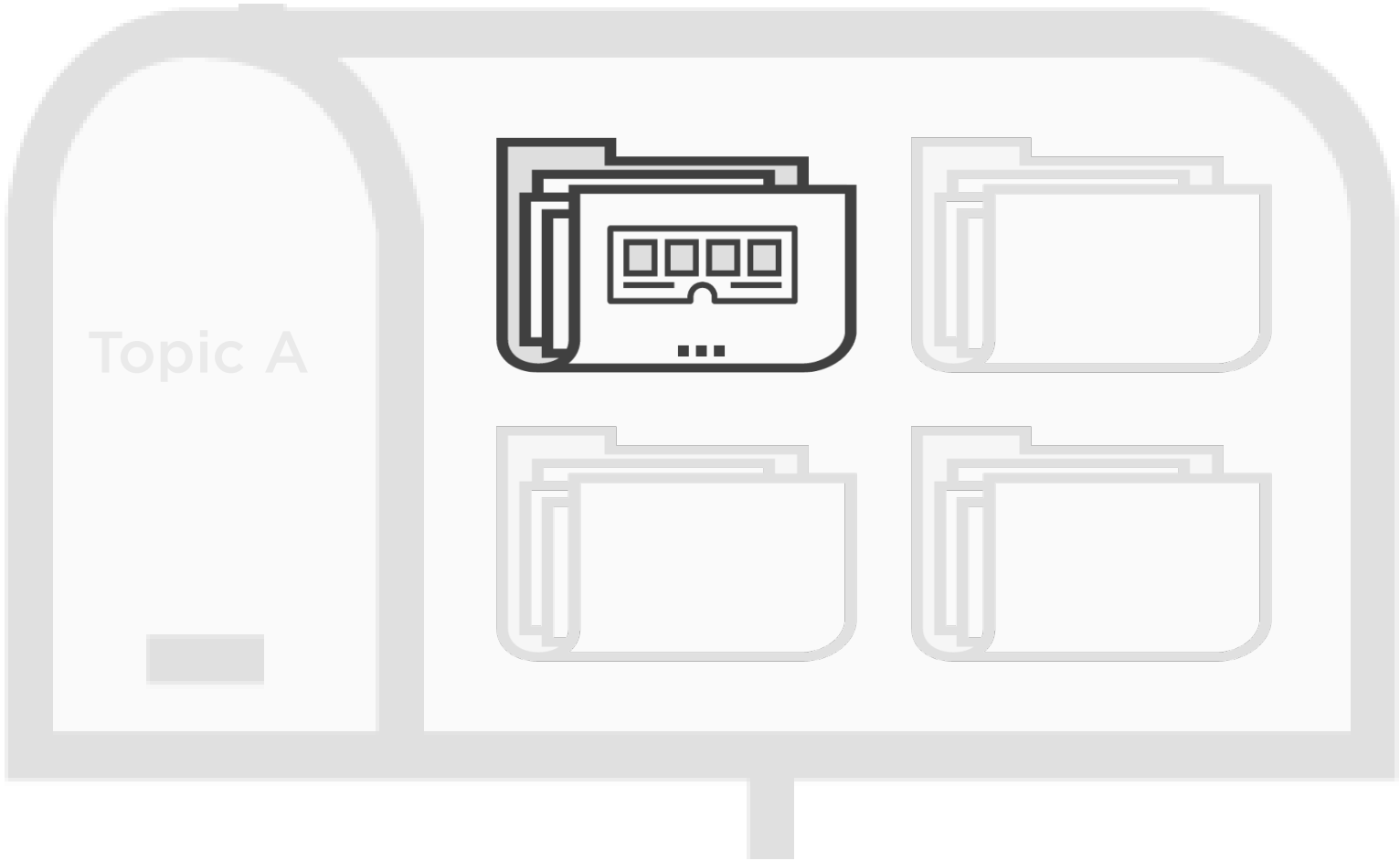


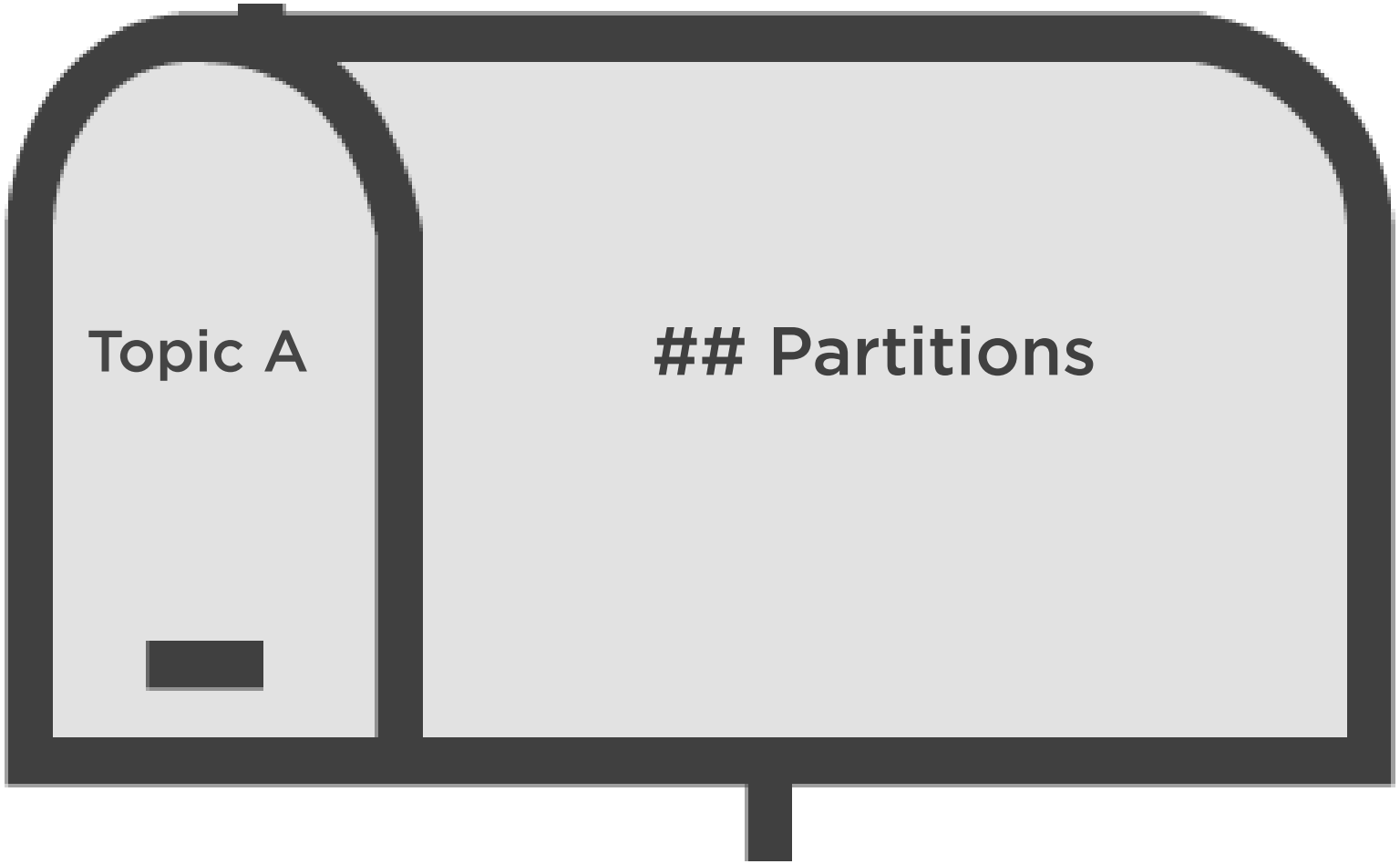
How Many Partitions?

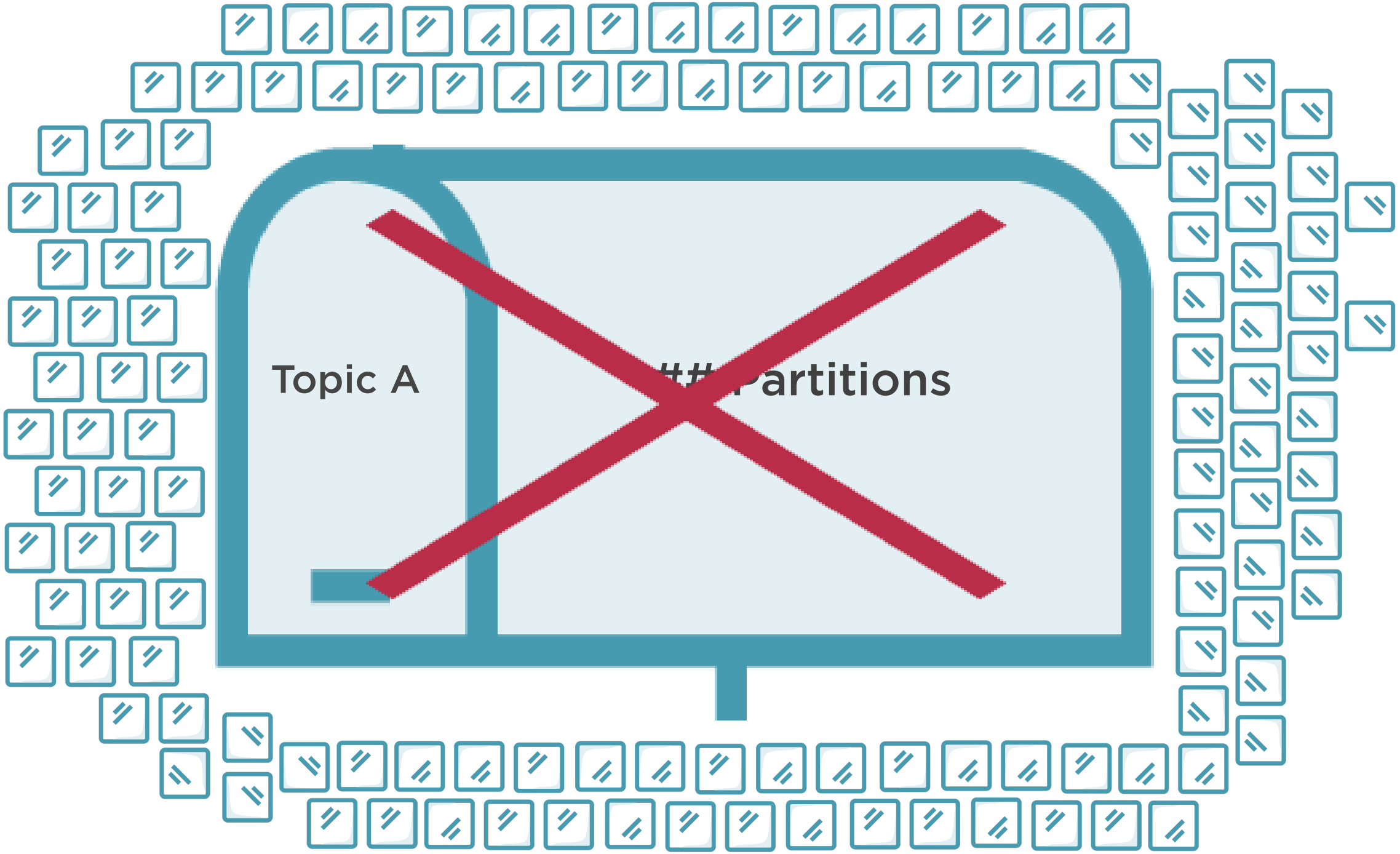


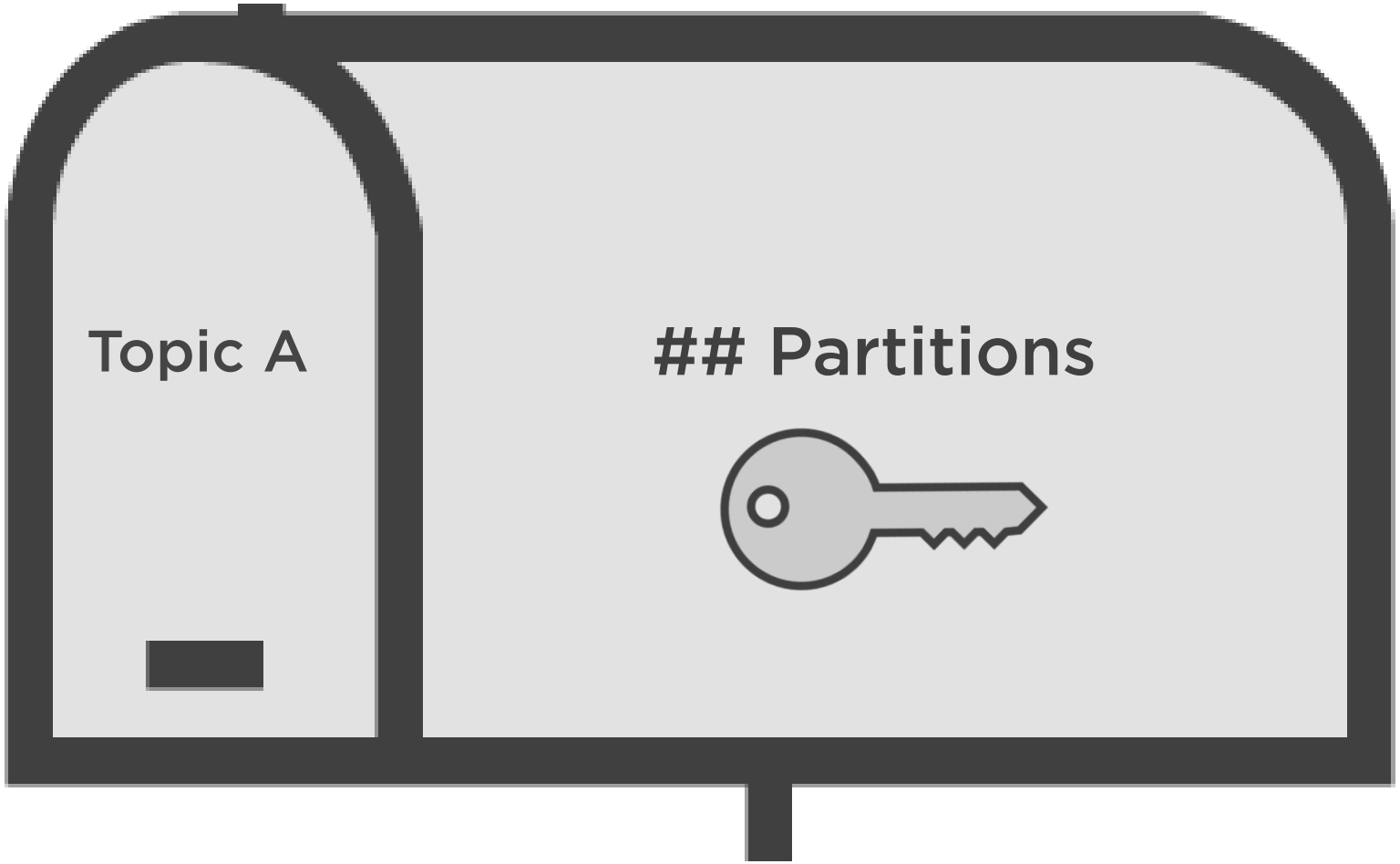


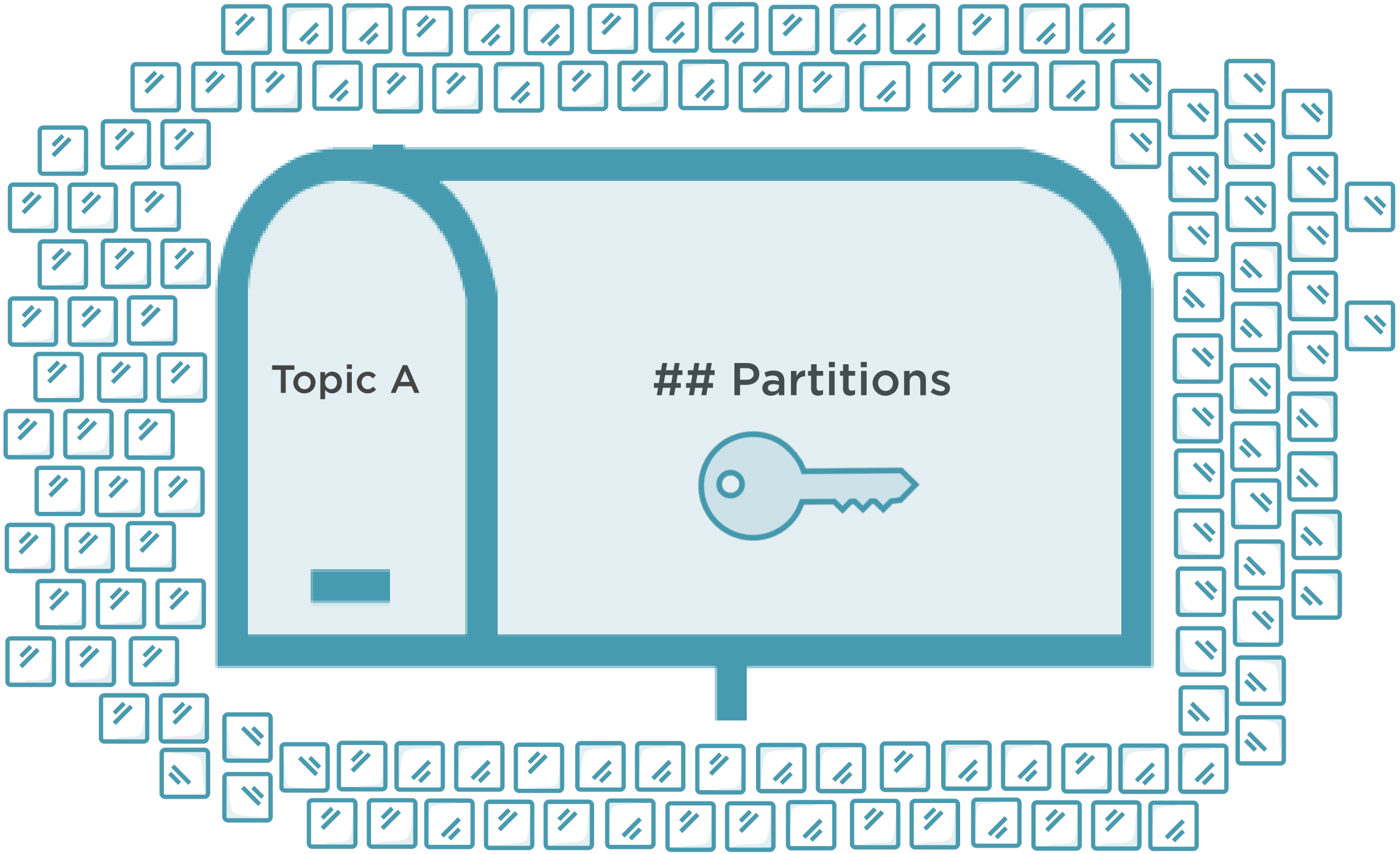
1/3

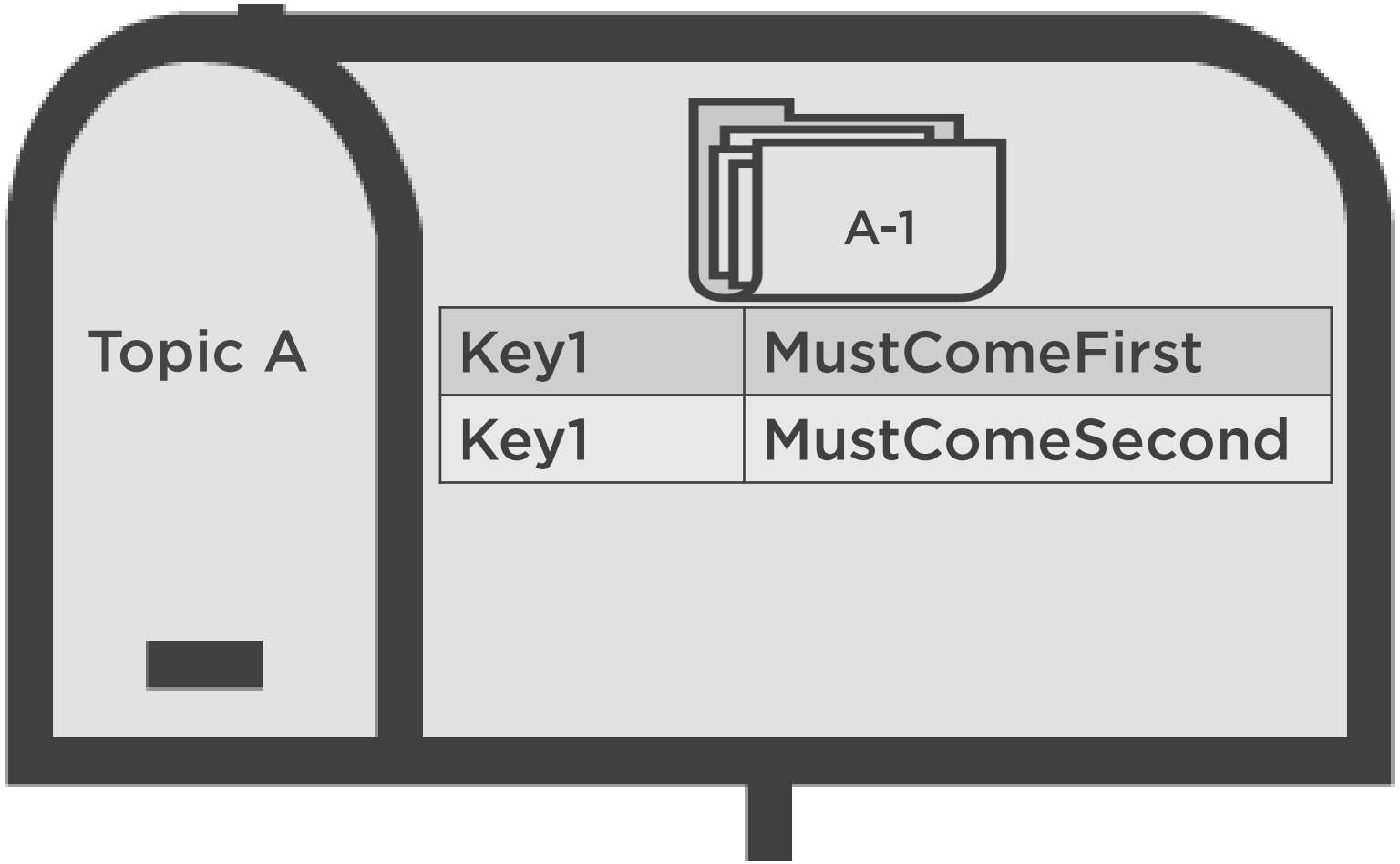

Topic A

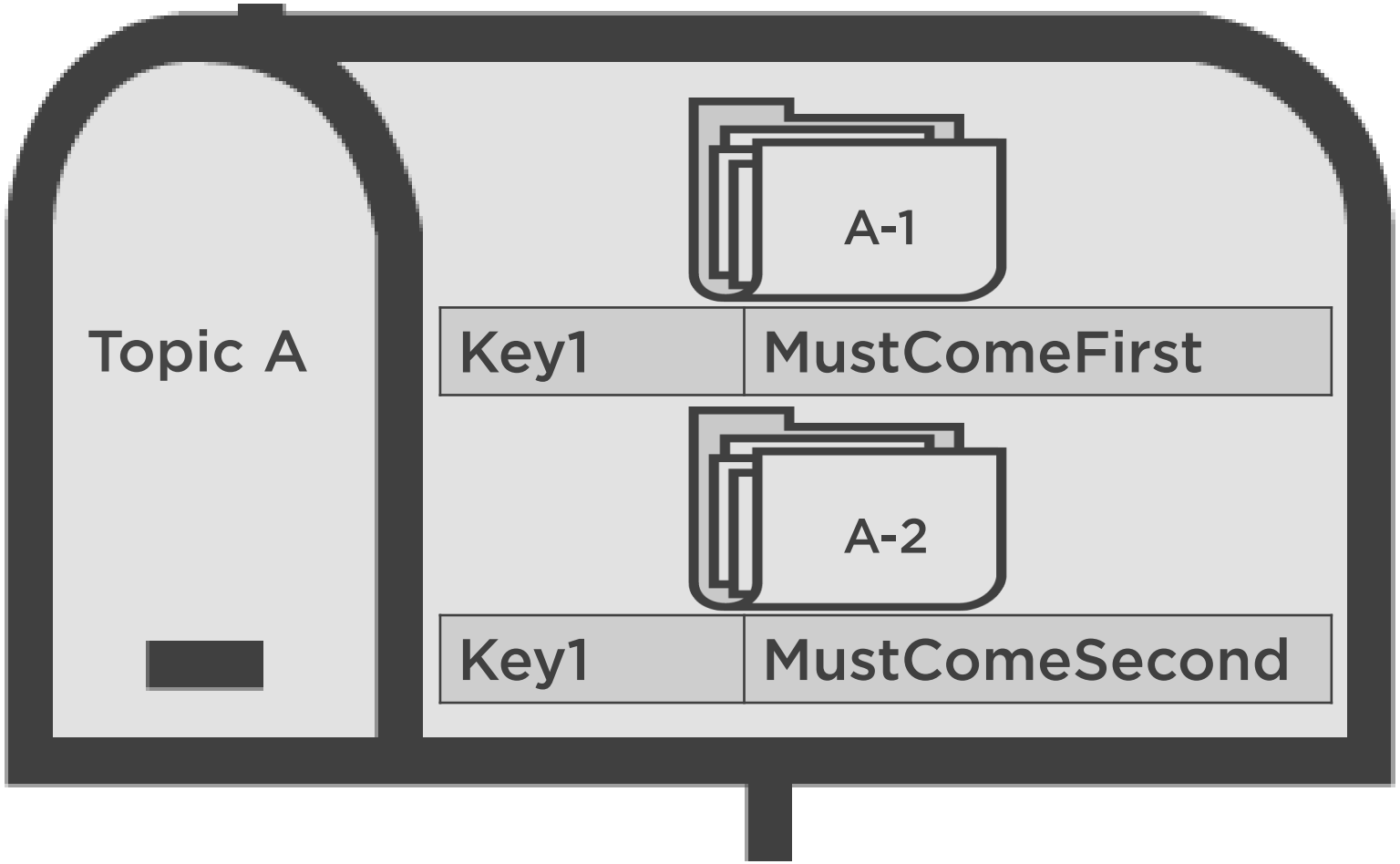

Topic A
Partitions

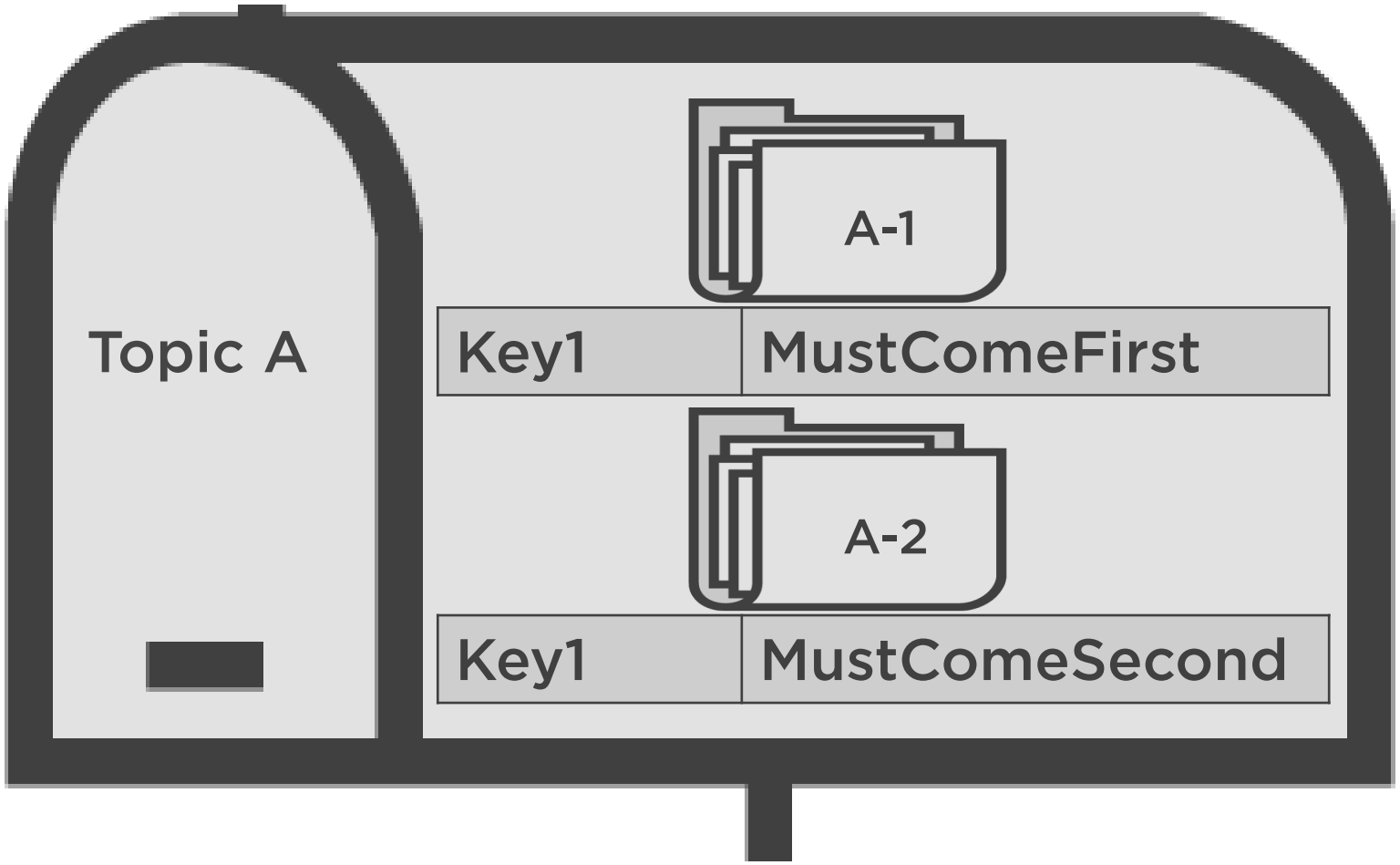

Topic A
Partitions


Topic A
Partitions

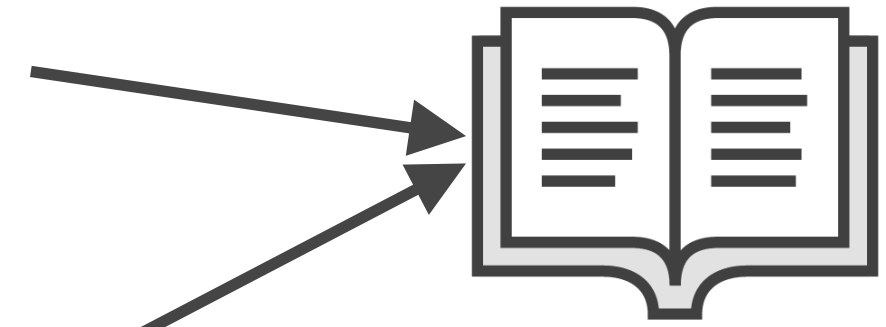

Topic A
Partitions


Topic A
A-1
Key1
MustComeFirst
Key1
MustComeSecond

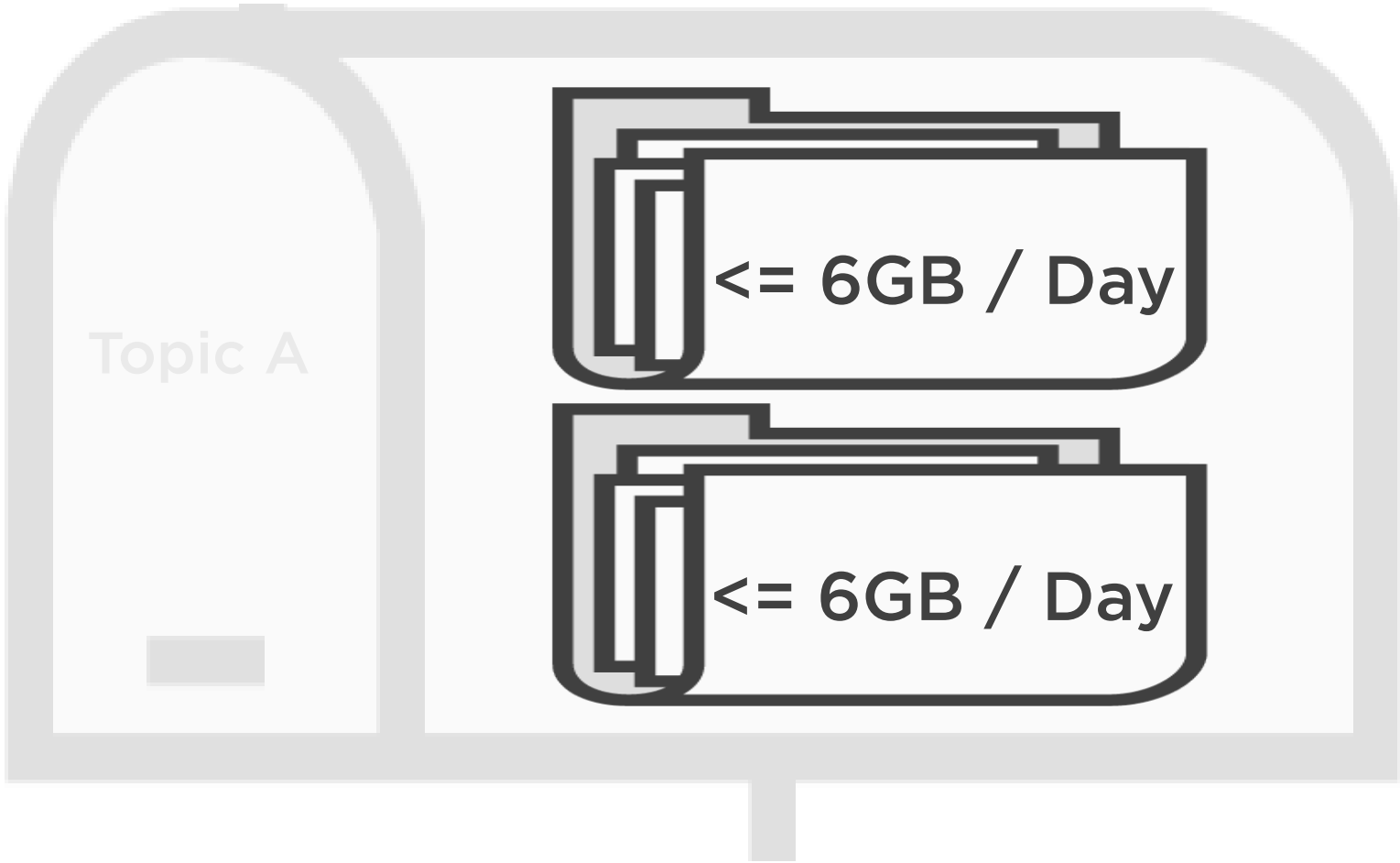

Topic A
A-1
Key1
MustComeFirst
A-2
Key1
MustComeSecond



CONSUMER



Key1	MustComeSecond
Key1	MustComeFirst


Topic A
<= 6GB / Day
<= 6GB / Day

Broker Config

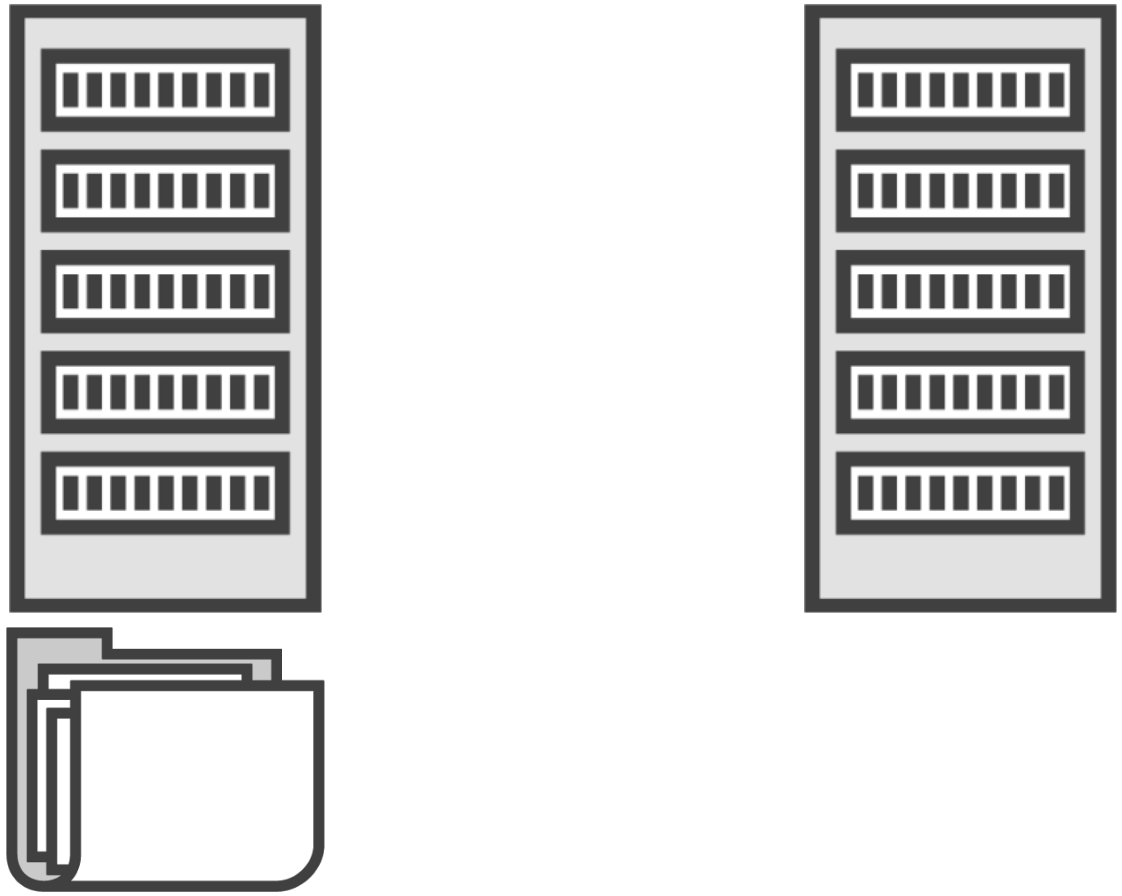
num.partitions (default = 1)

kafka-topics.sh --create ... **--partitions ##**

auto.create.topics.enable (default = true)

Configuring the Broker: Fault Tolerance via Replication

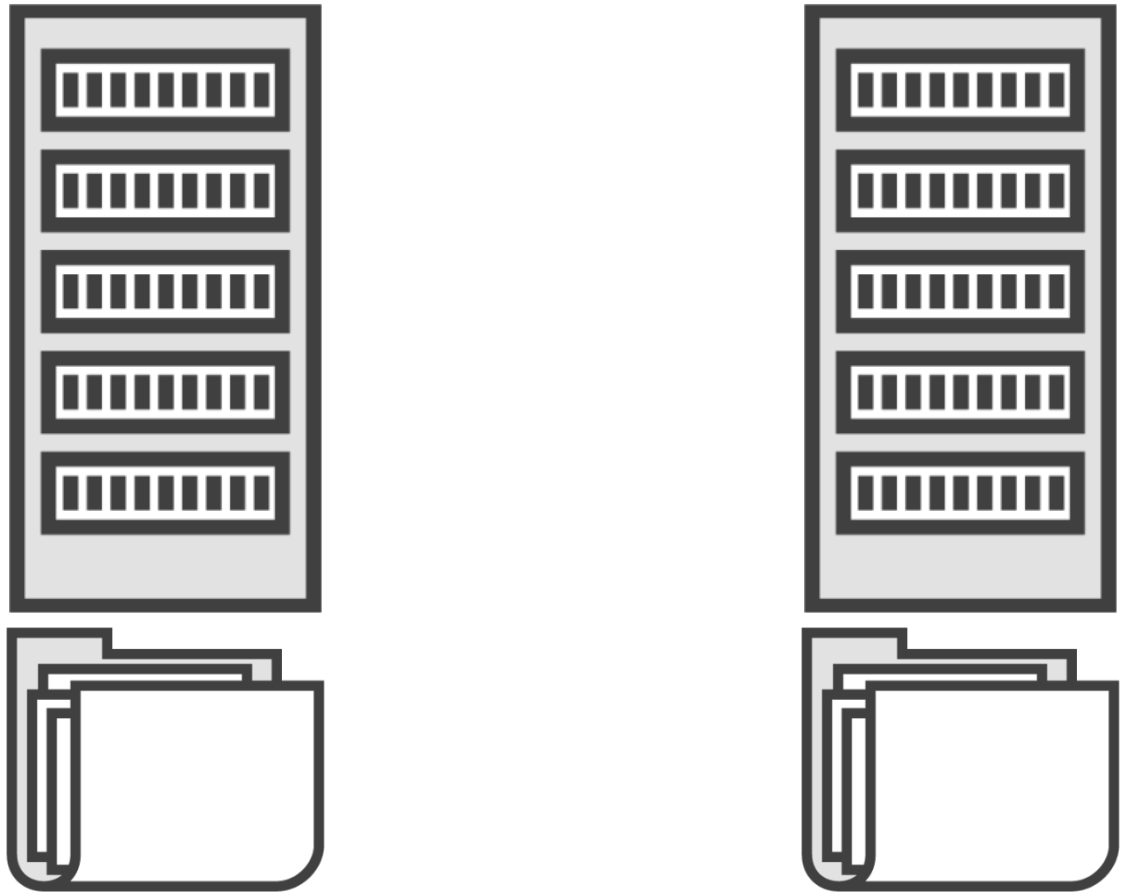
Replication



Replication

kafka-topics.sh --create ... **--replication-factor ##**

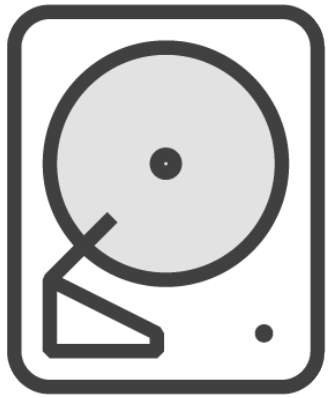
Replication



replication-factor = 2

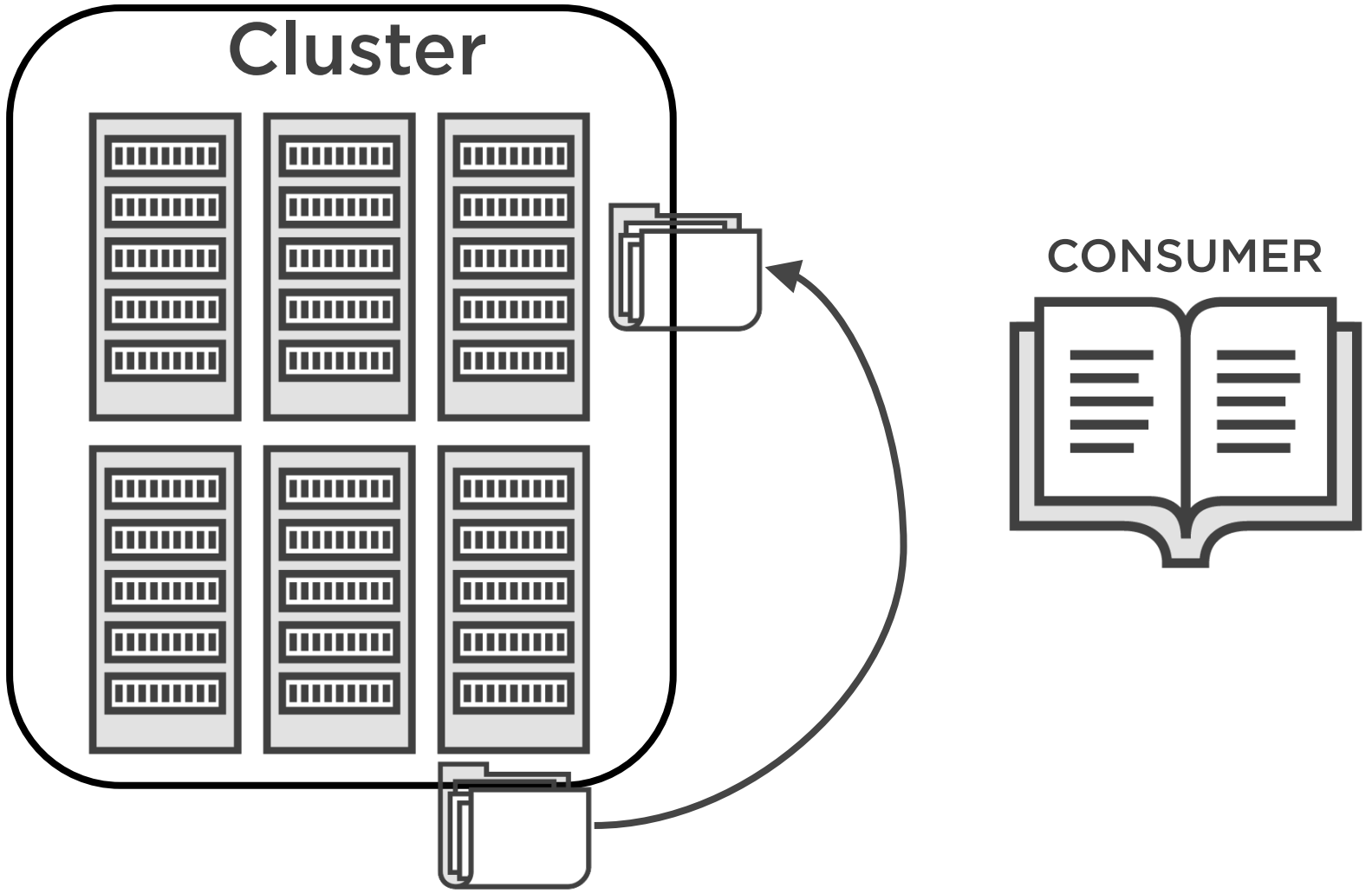
What to Set Replication Factor?

What to Set Replication Factor?



What to Set Replication Factor?

What to Set Replication Factor?



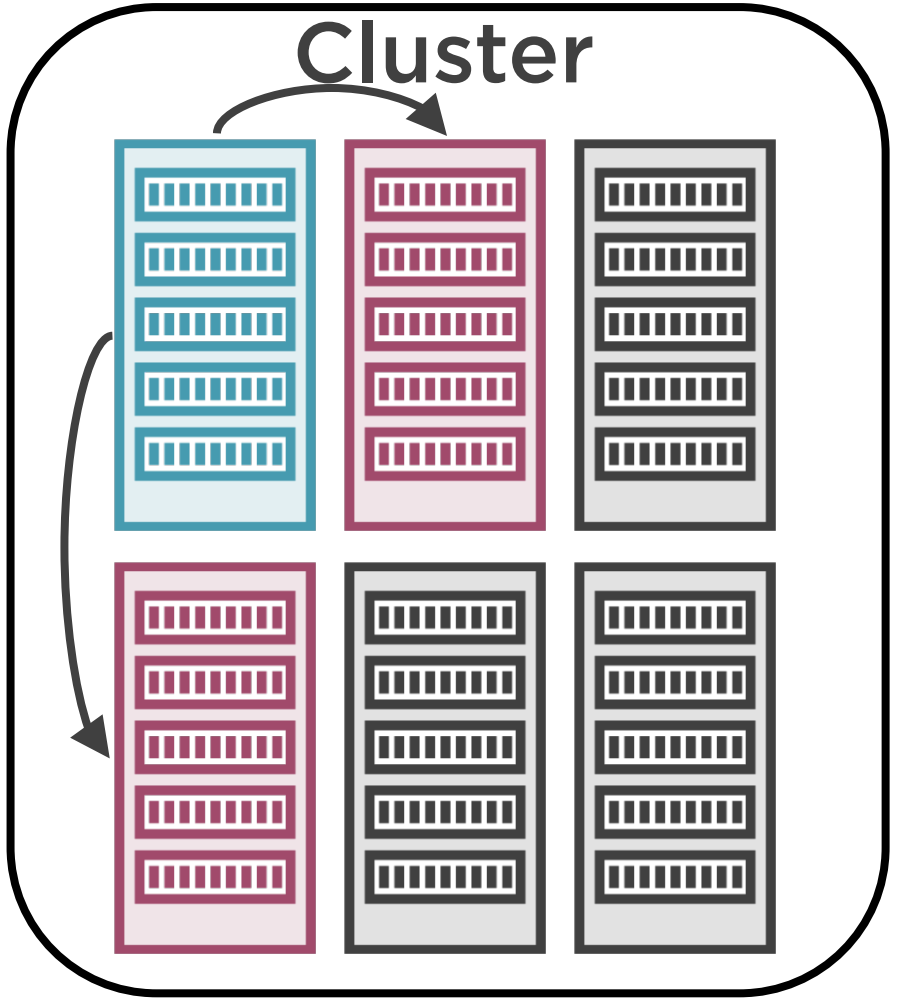
What to Set Replication Factor?

Replication

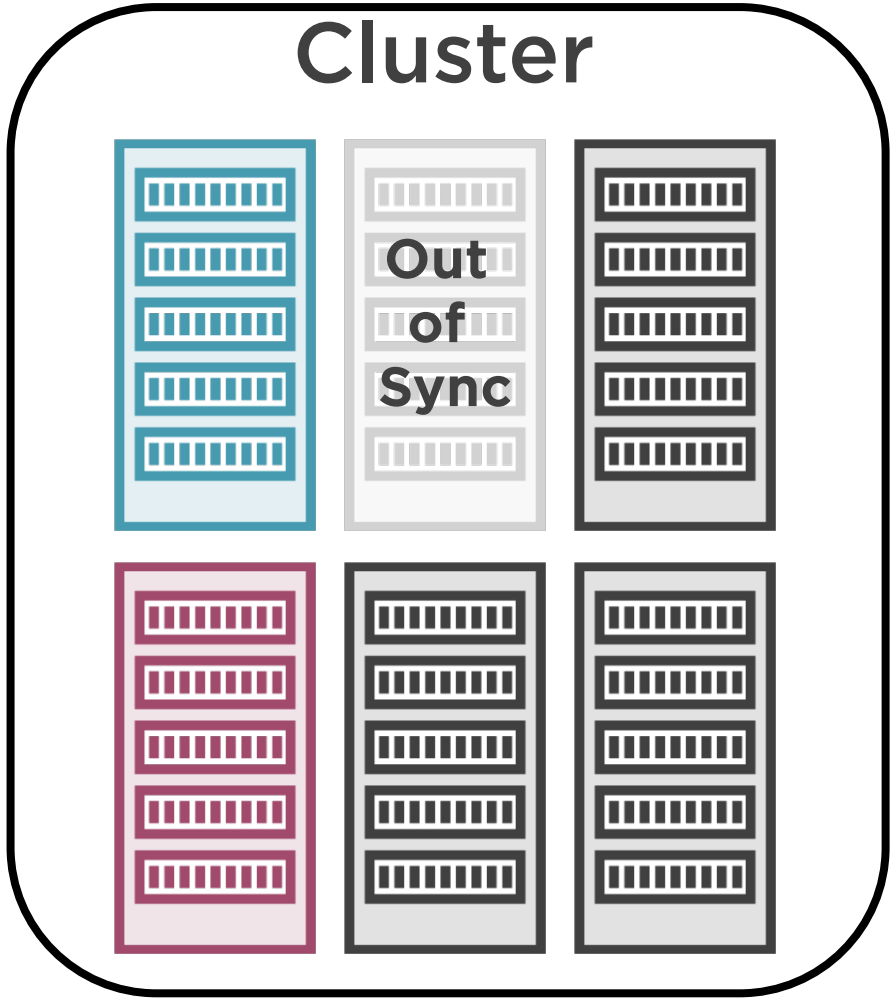
kafka-topics.sh --create ... **--replication-factor ##**

default.replication.factor (default = 1)
(only for automatic topic creation)

Replication



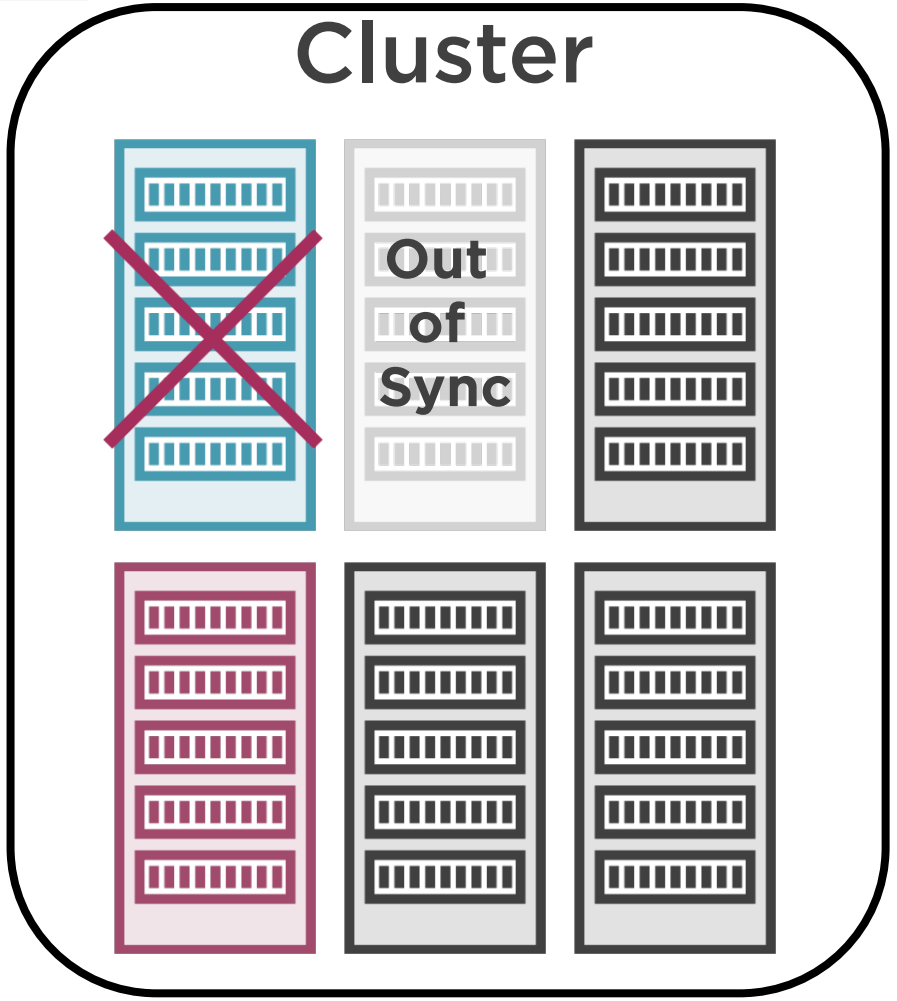
Replication



unclean.leader.election.enable

>= Kafka 0.11 (default = false)
<= Kafka 0.11 (default = true)

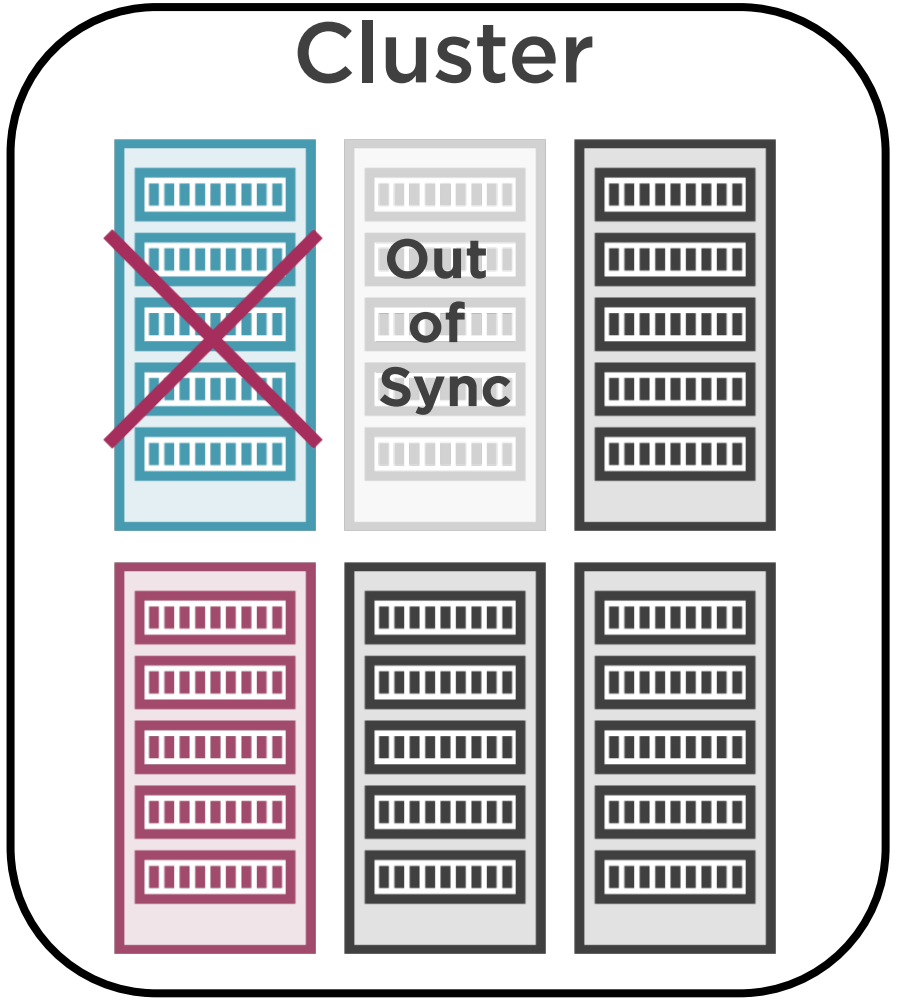
Replication



unclean.leader.election.enable

true

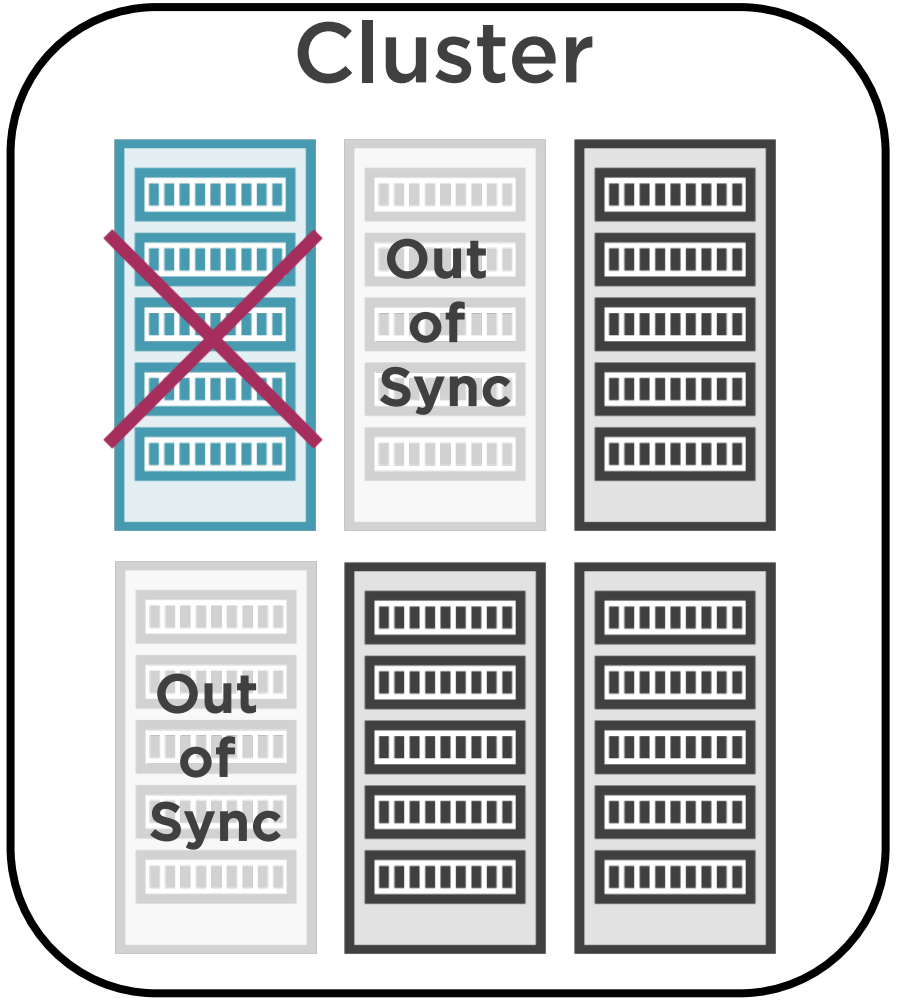
Replication



unclean.leader.election.enable

true

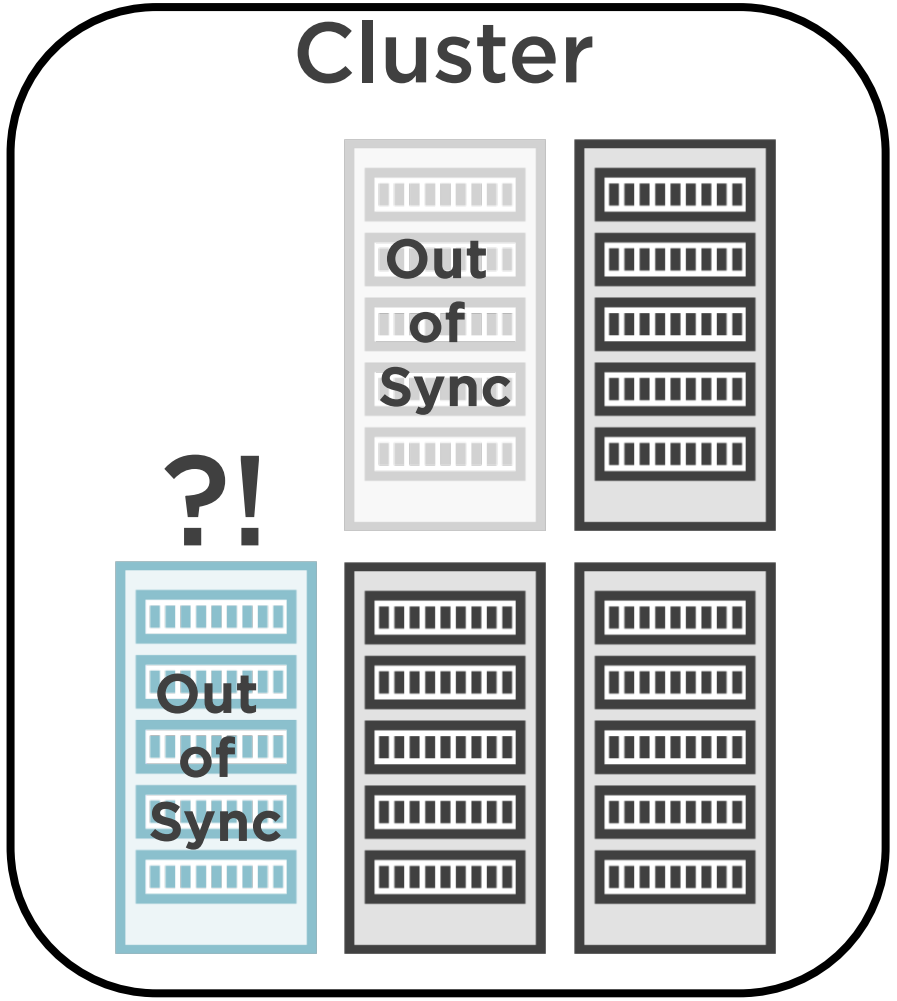
Replication



unclean.leader.election.enable

true

Replication

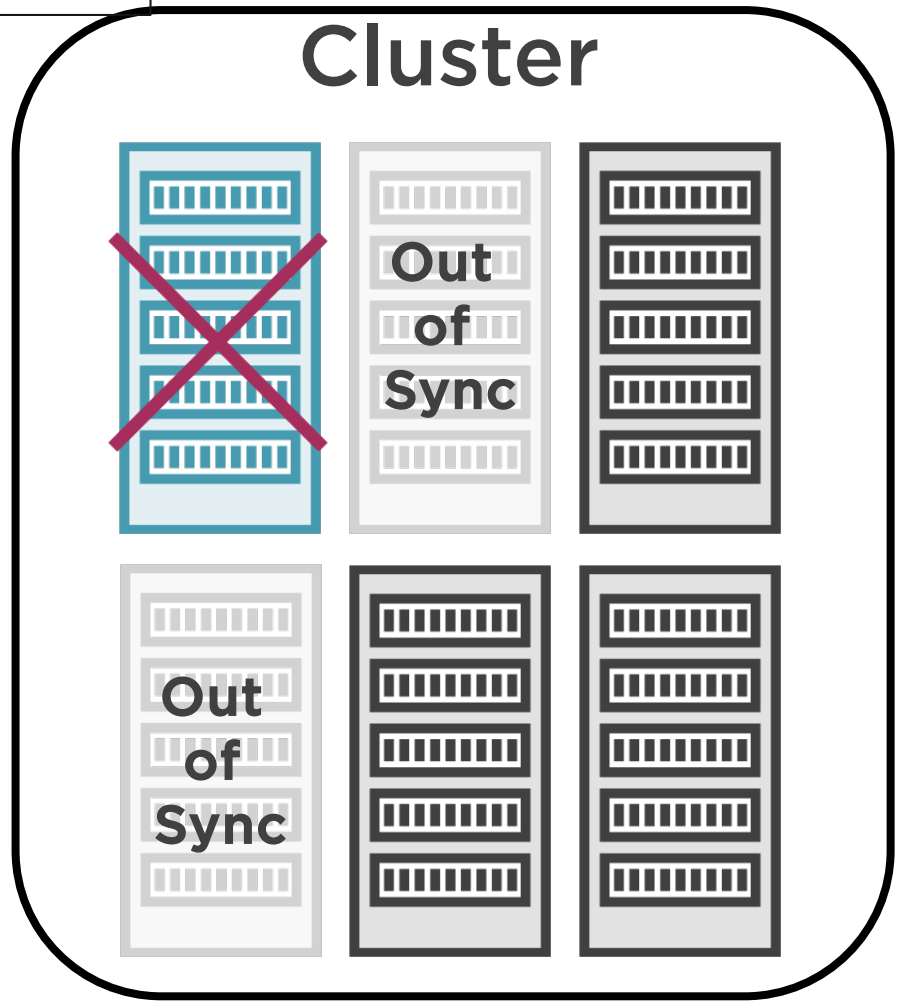


Replication

unclean.leader.election.enable

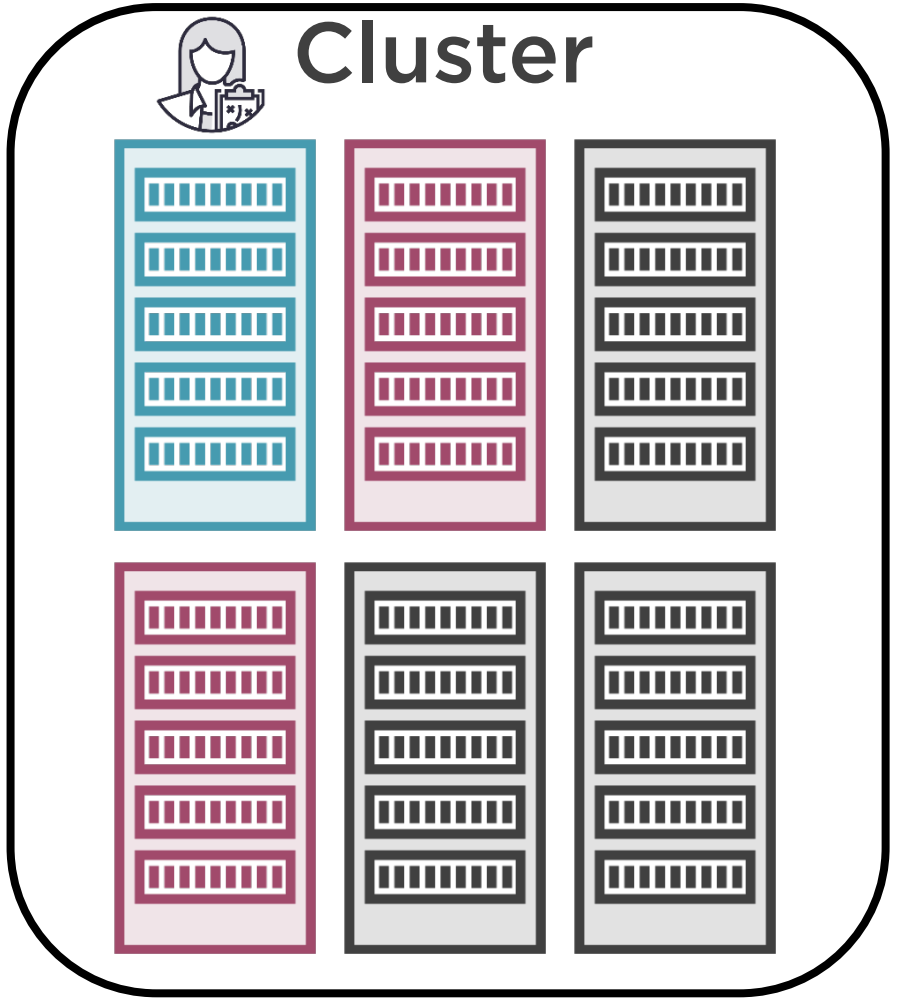
>= Kafka 0.11 (default = false)

<= Kafka 0.11 (default = true)

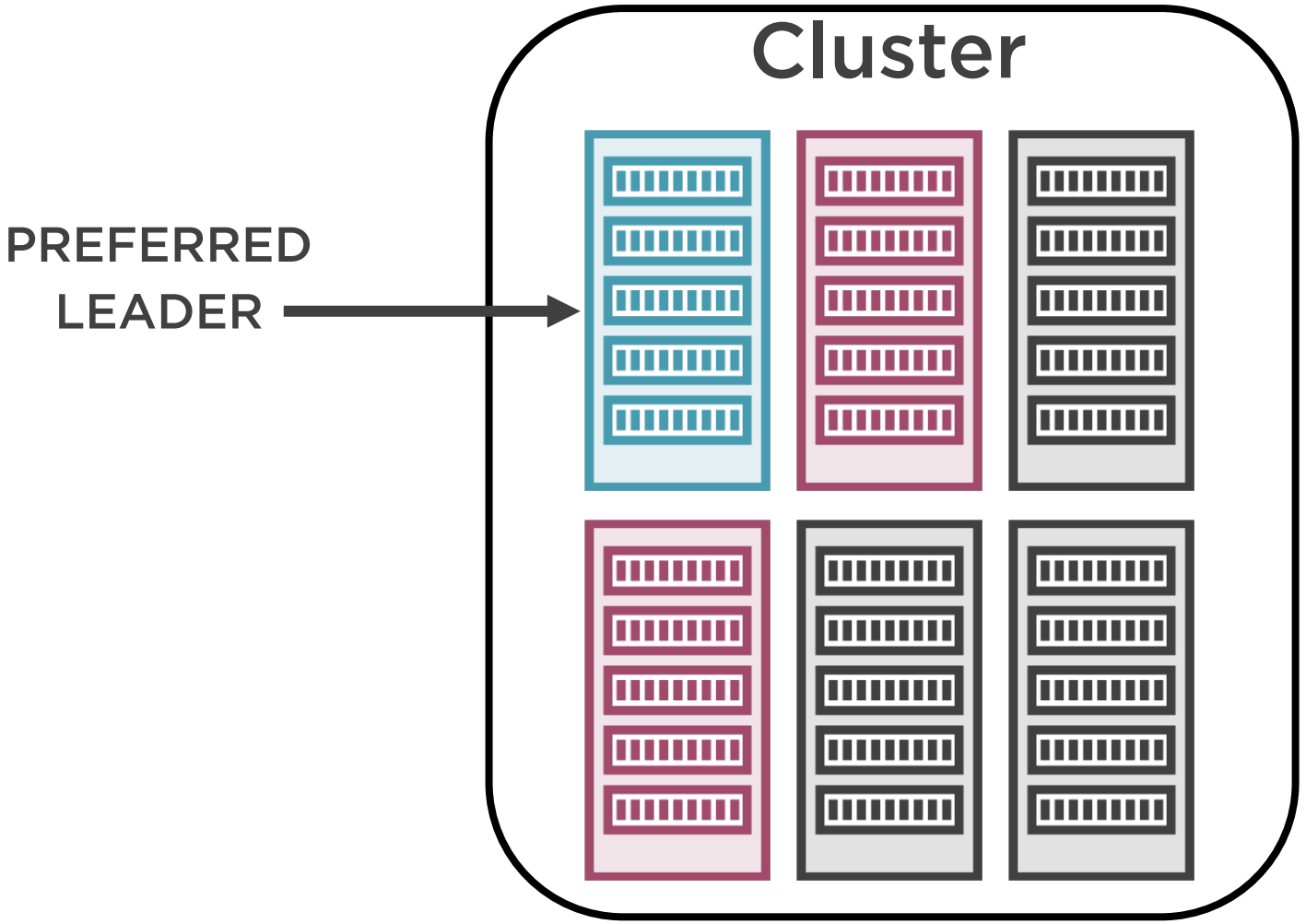


Replication

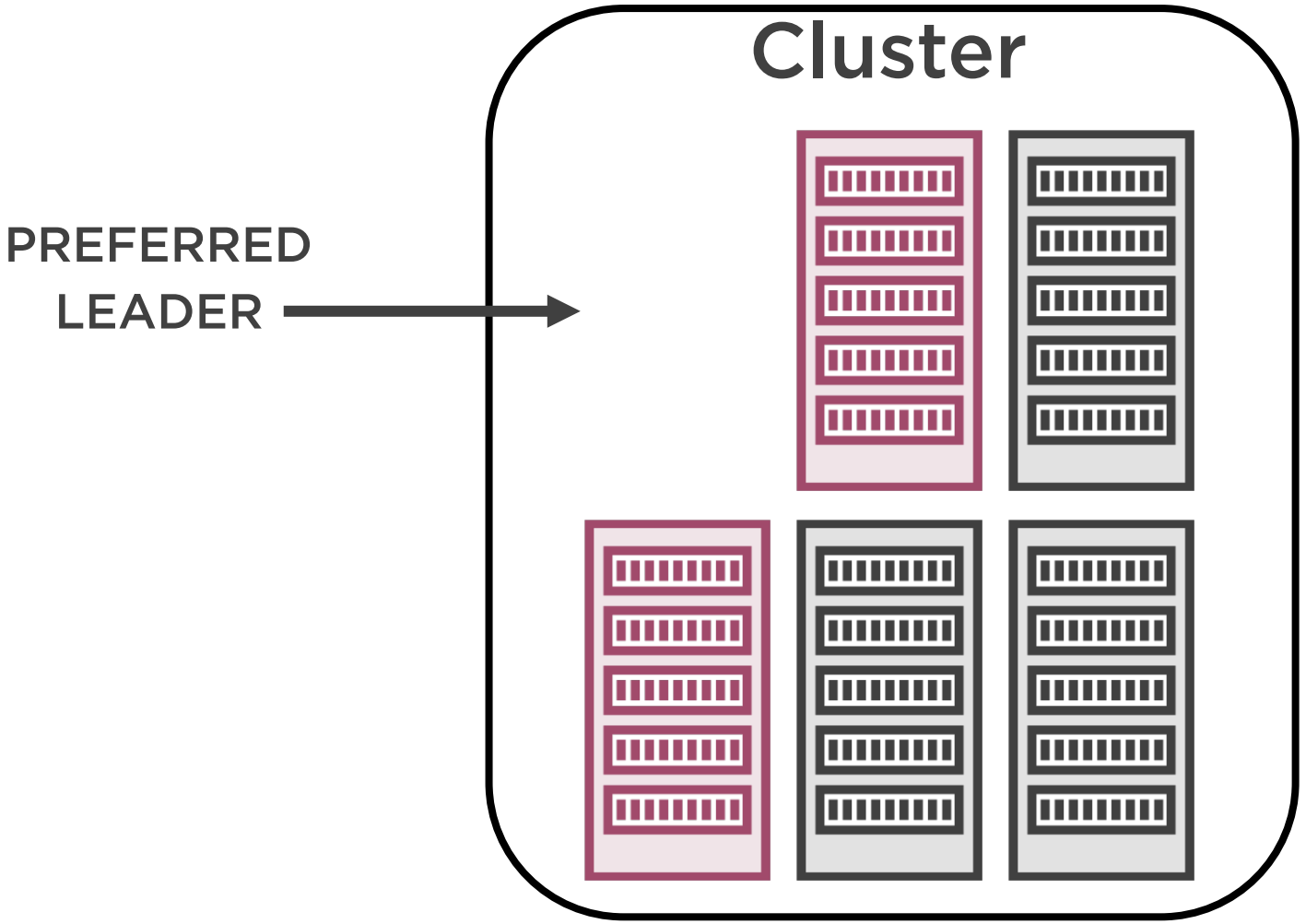
replica.lag.time.max.ms
10000



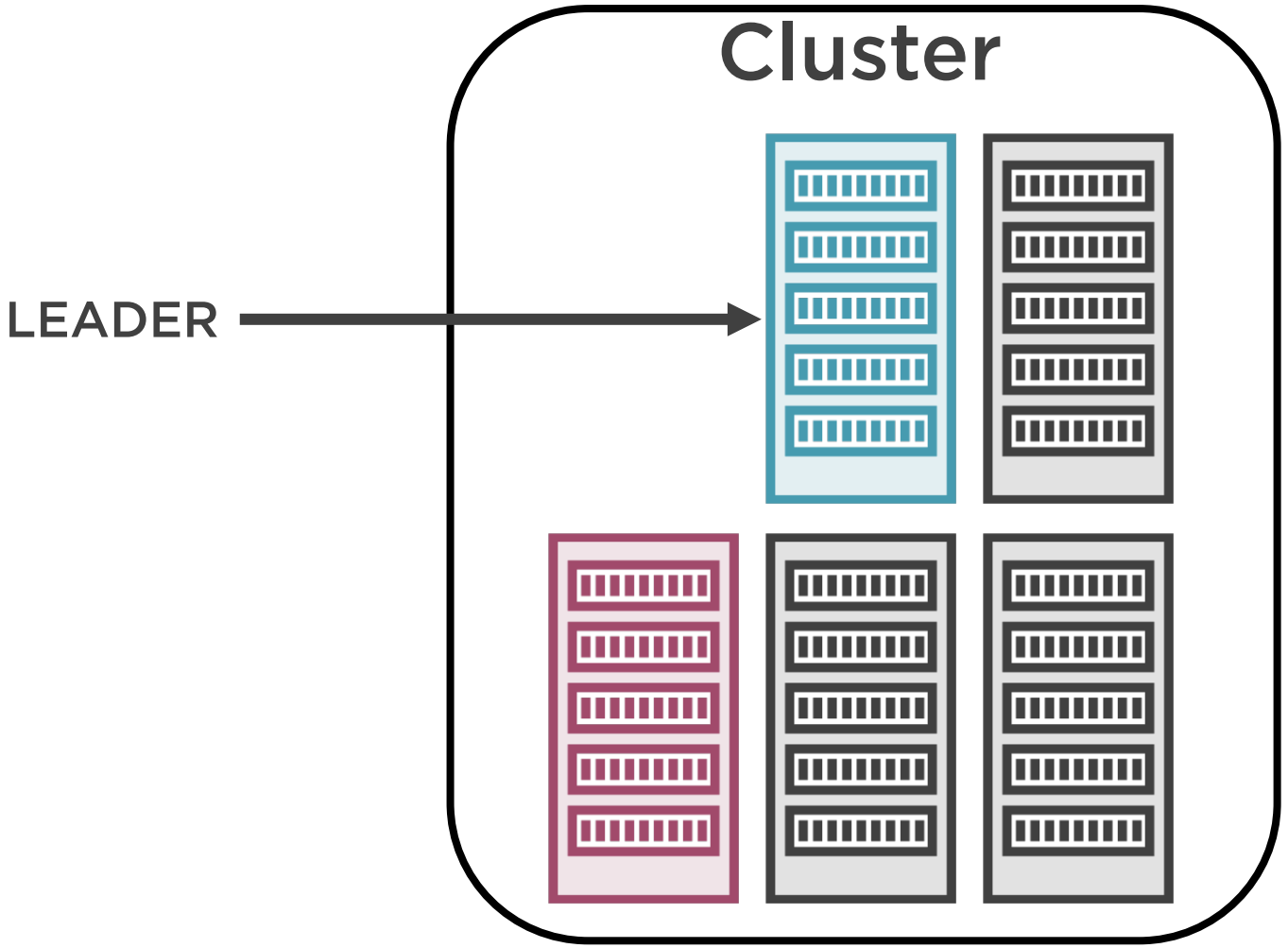
Replication



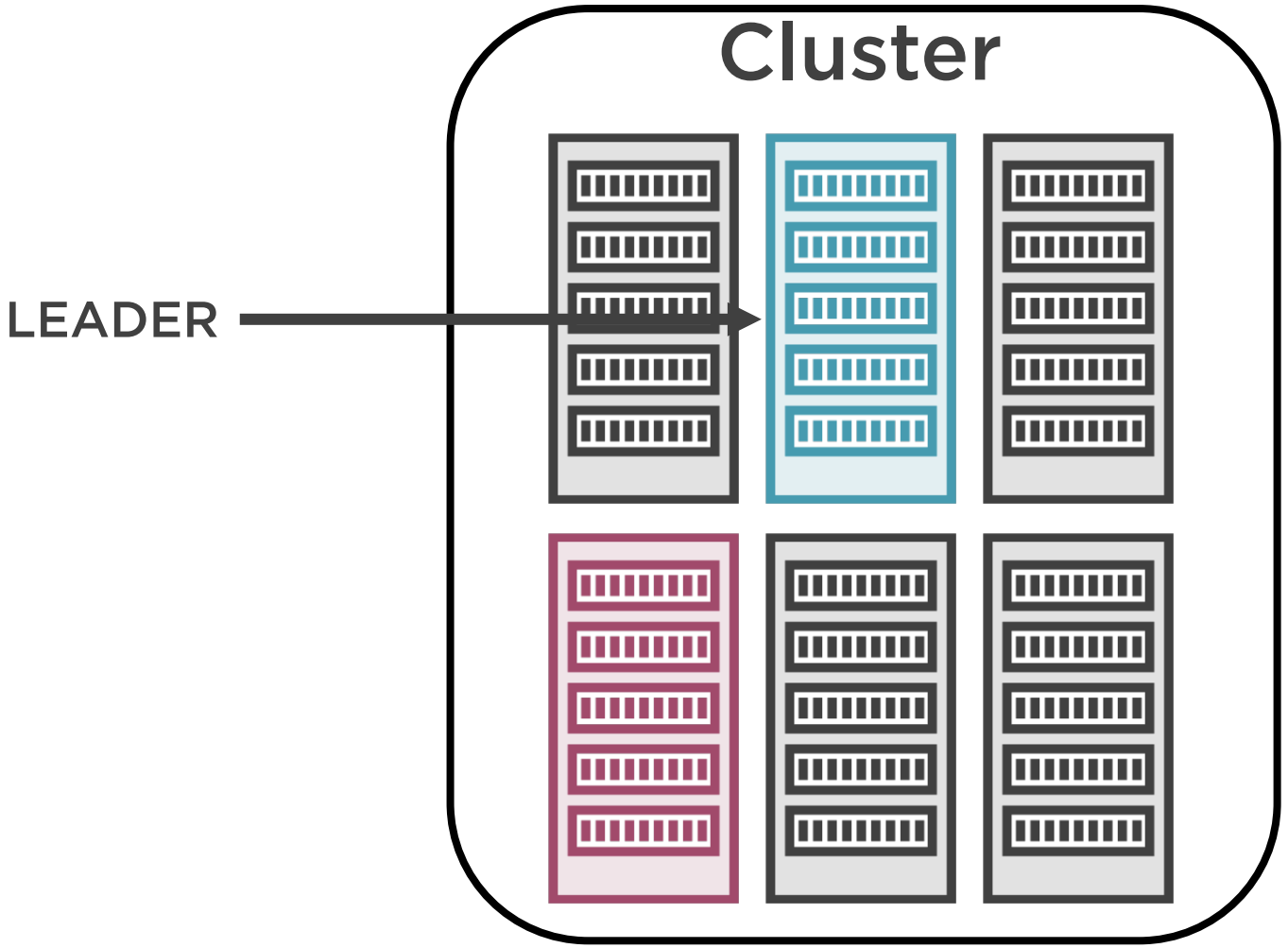
Replication



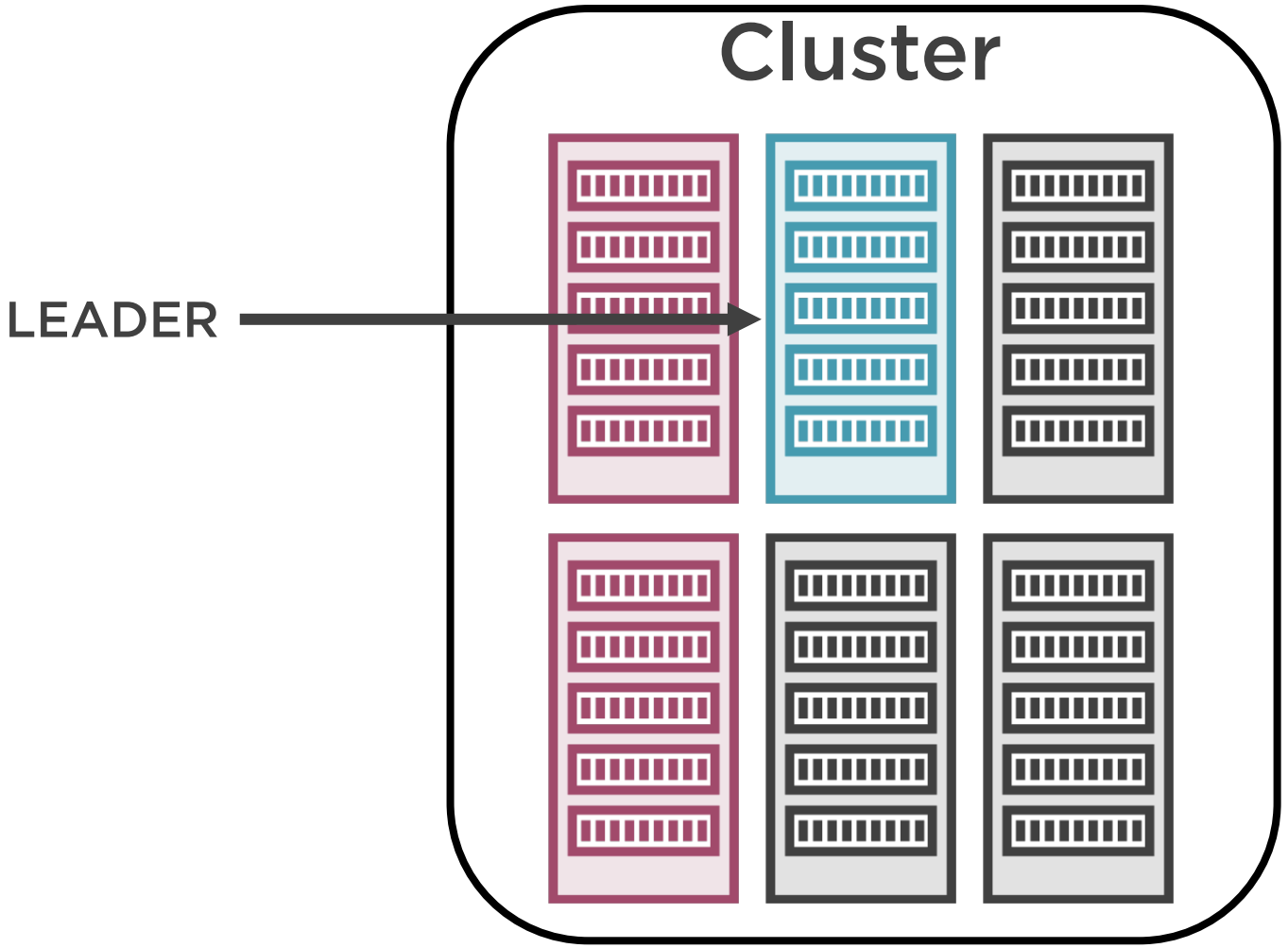
Replication



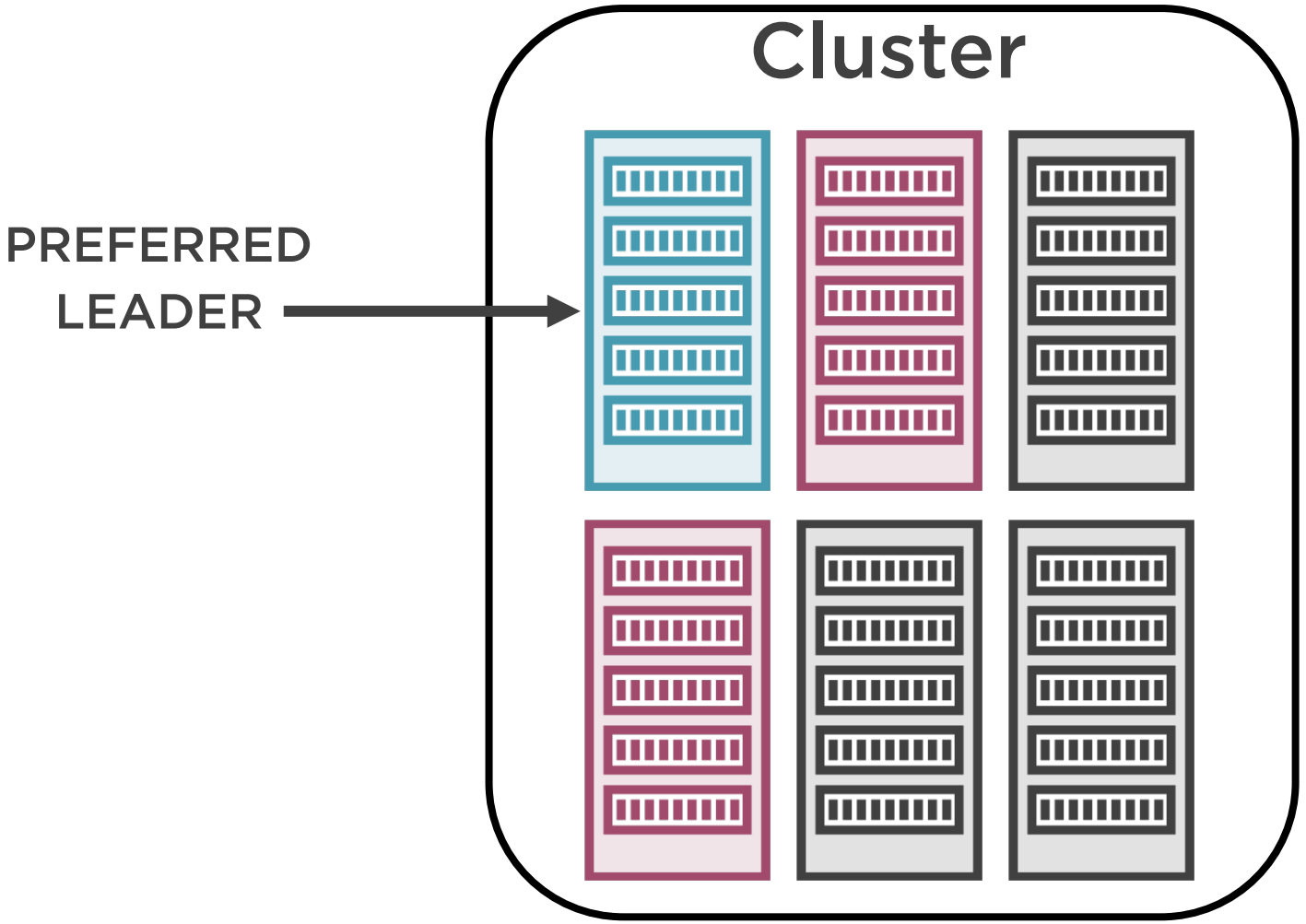
Replication



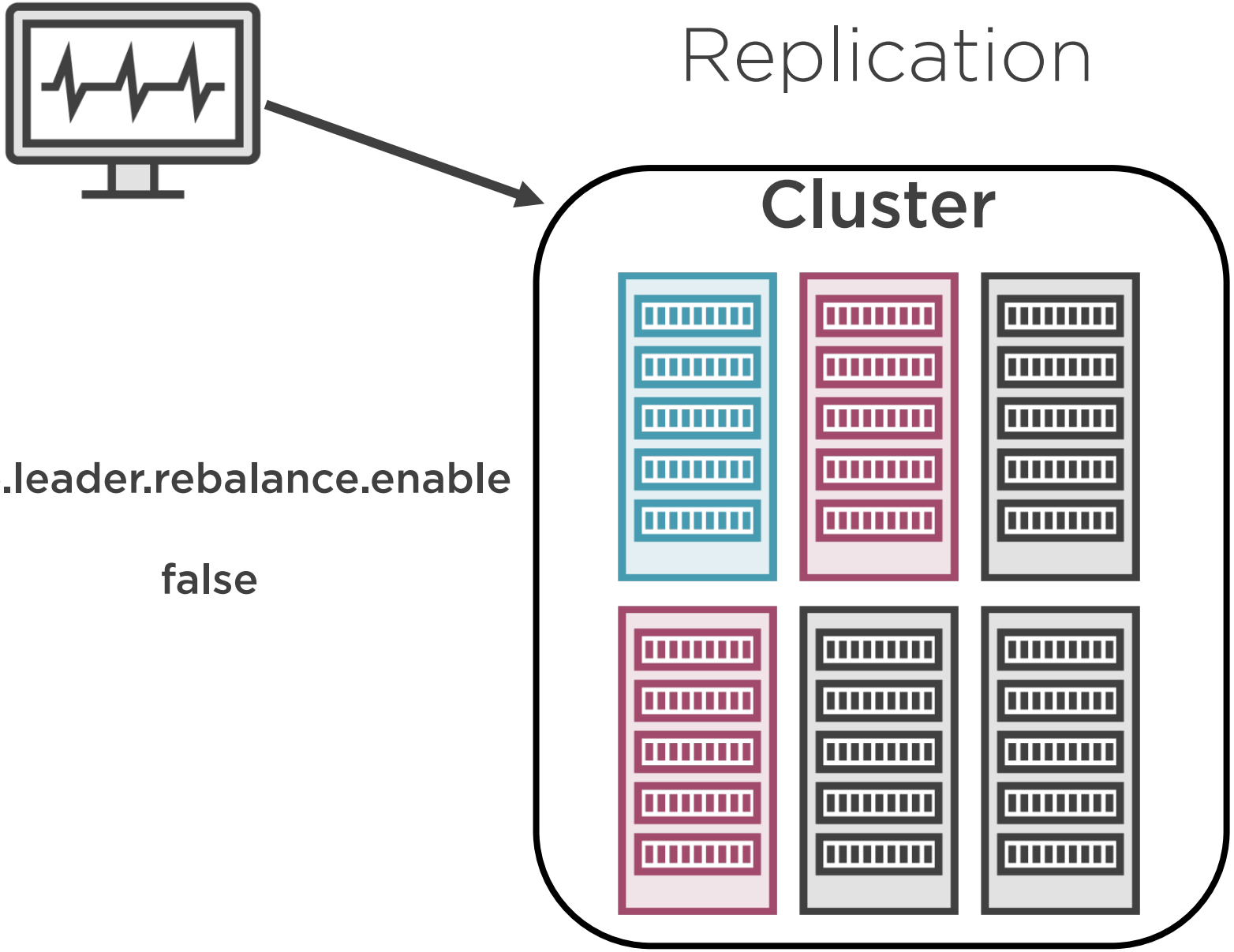
Replication



Replication



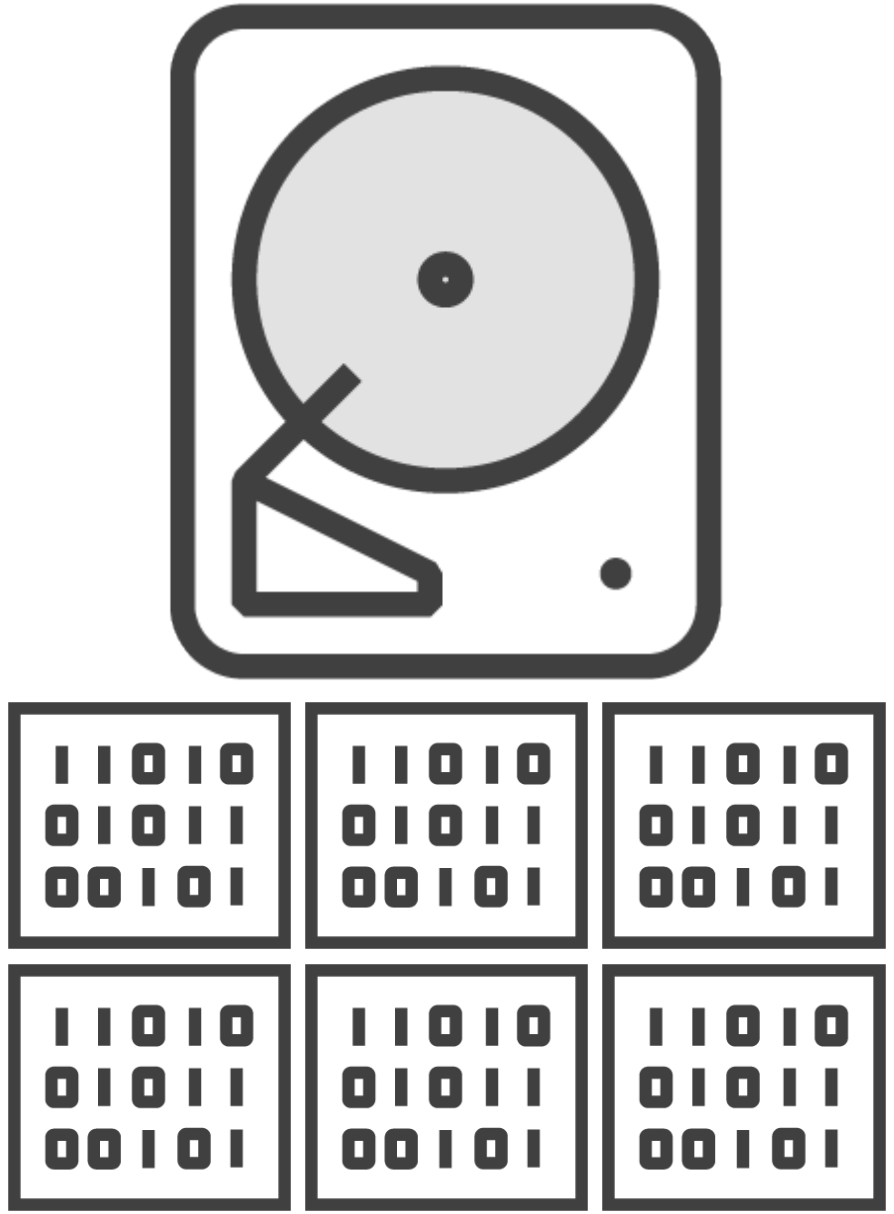
Replication

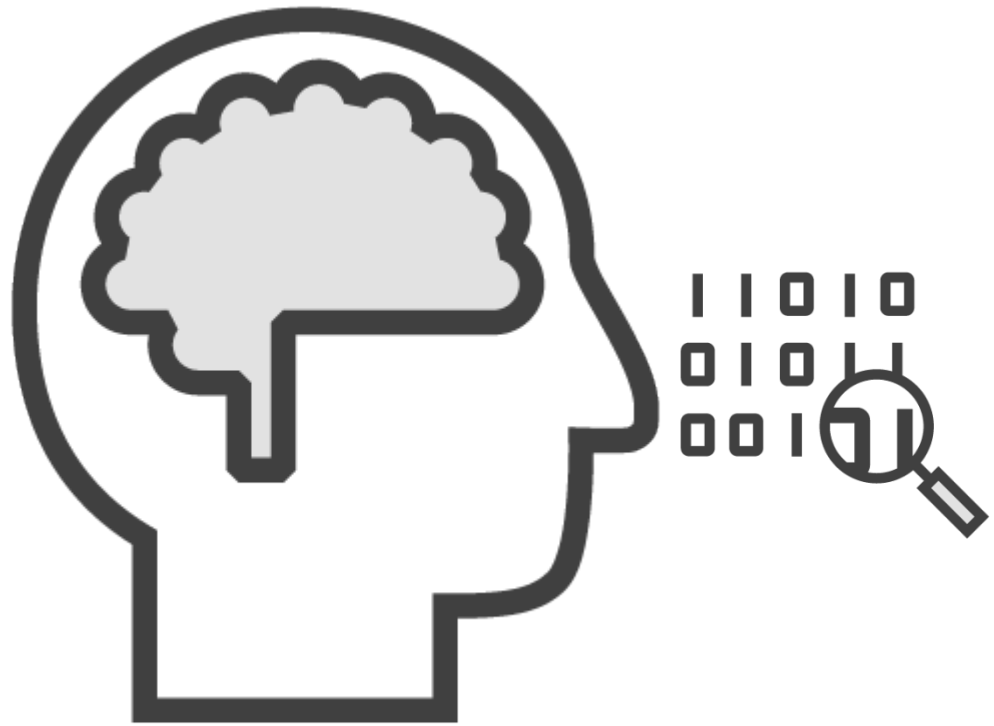


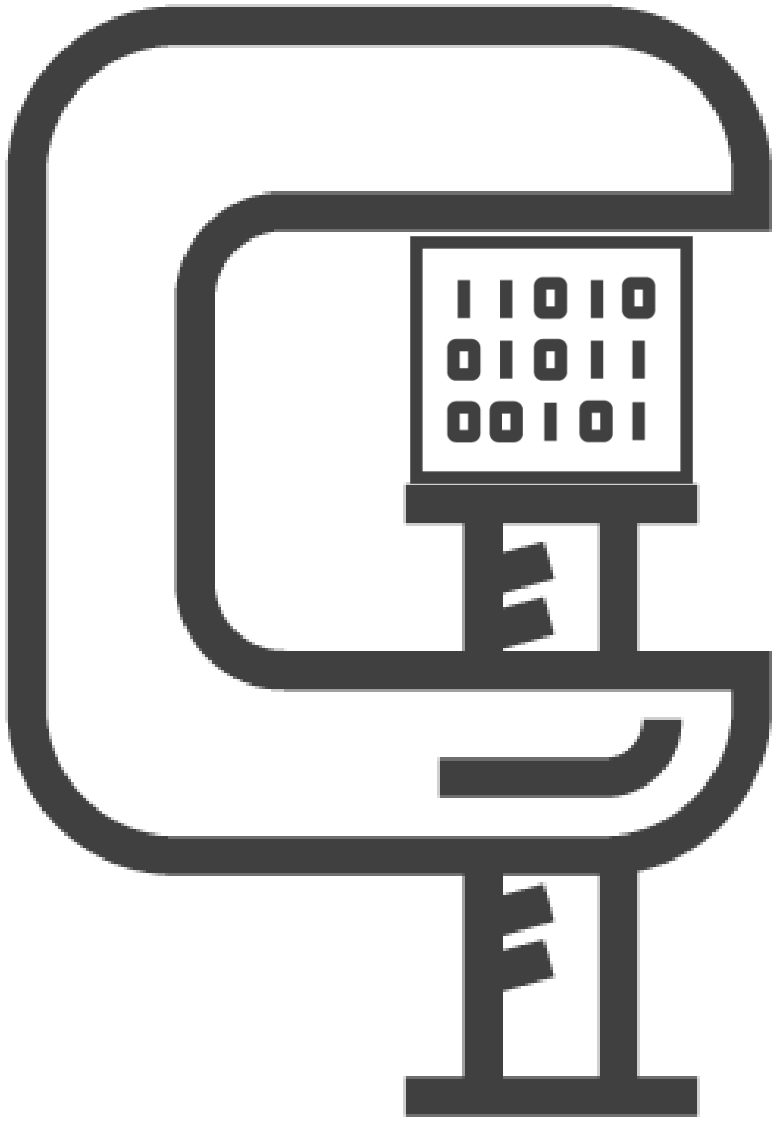
auto.leader.rebalance.enable

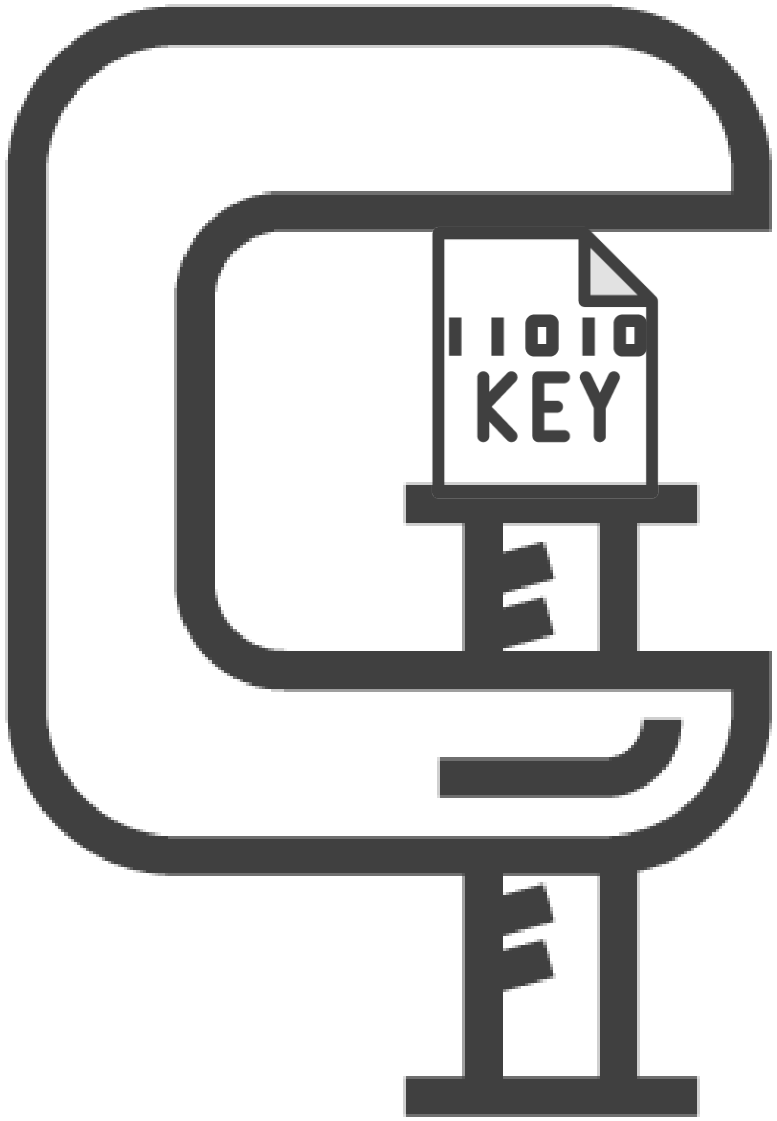
false

Configuring the Broker: Data Retention





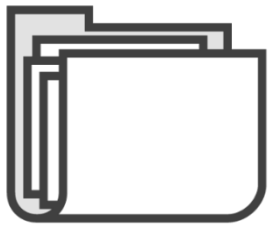

11010
01011
00101


11010
KEY

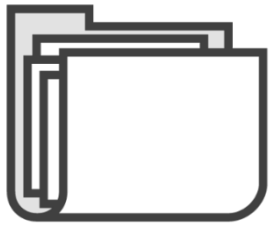
Data Management

log.dirs

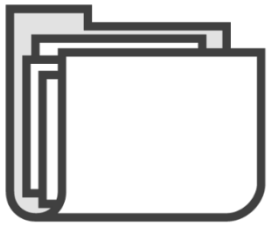
(server.properties default = tmp/kafka-logs)



5 Partitions



7 Partitions



3 Partitions

0 Partitions

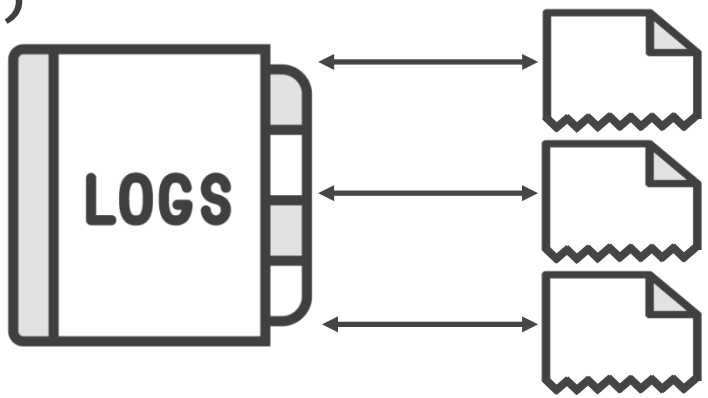
Data Management

log.dirs

(server.properties default = tmp/kafka-logs)

num.recovery.threads.per.data.dir

(default = 1)



Data Management

log.dirs

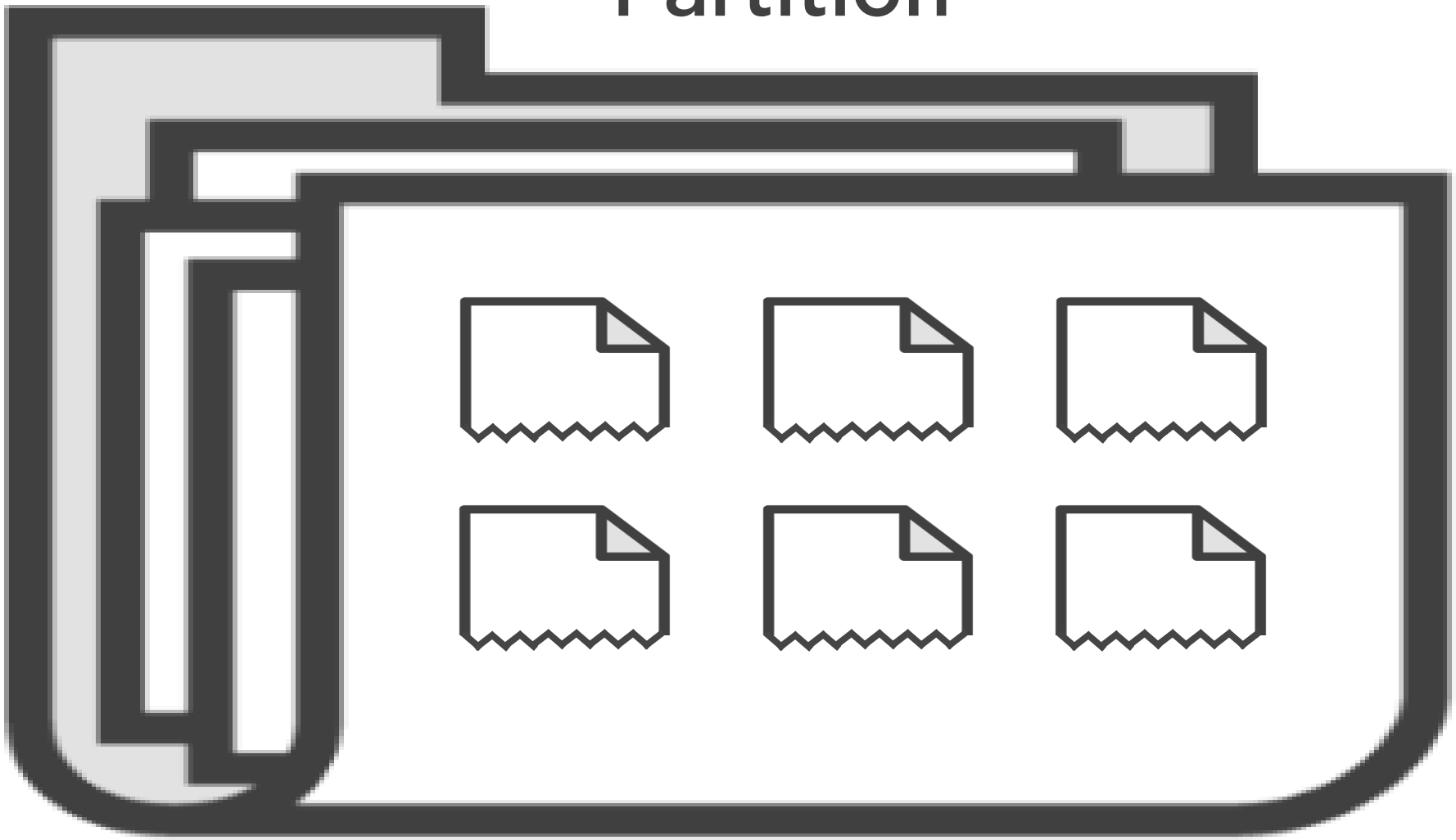
(server.properties default = tmp/kafka-logs)

num.recovery.threads.per.data.dir

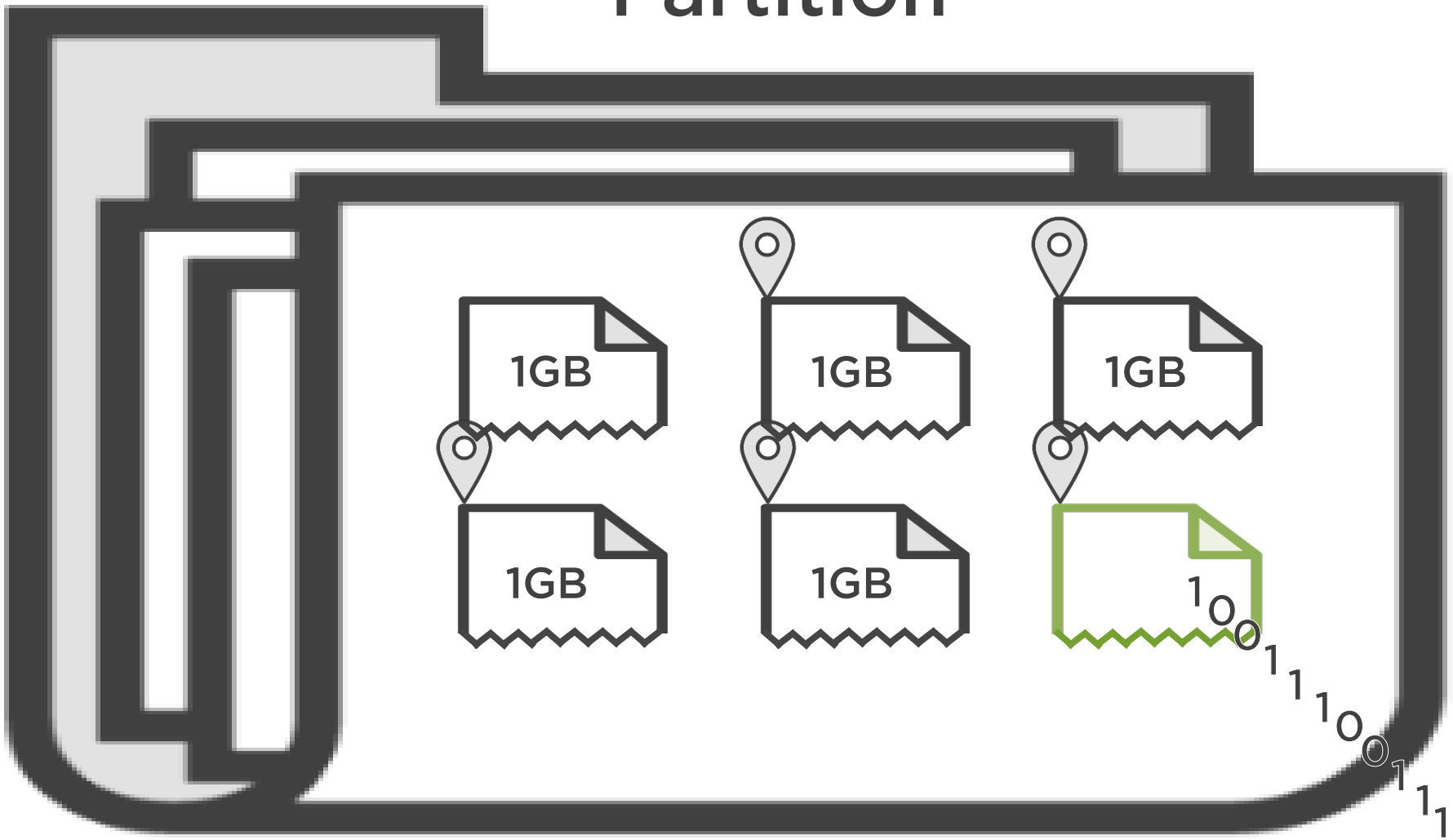
(default = 1)

log.dirs
x
recovery

Partition



Partition



log.segment.bytes (default = 1073741824 (1GB))

log.retention.bytes

log.retention.[hours/minutes/ms]

log.retention.bytes (per partition)

(default = -1 (unlimited))

log.retention.[hours/minutes/ms]

# of partions
x
retention

log.retention.bytes (per partition)

(default = -1 (unlimited))

log.retention.[hours/minutes/ms] (per partition)

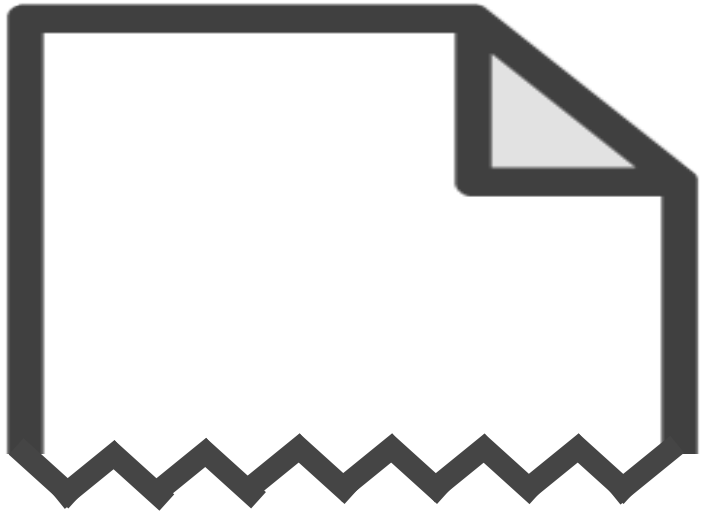
(default = 168 hours)

log.retention.bytes (per partition)

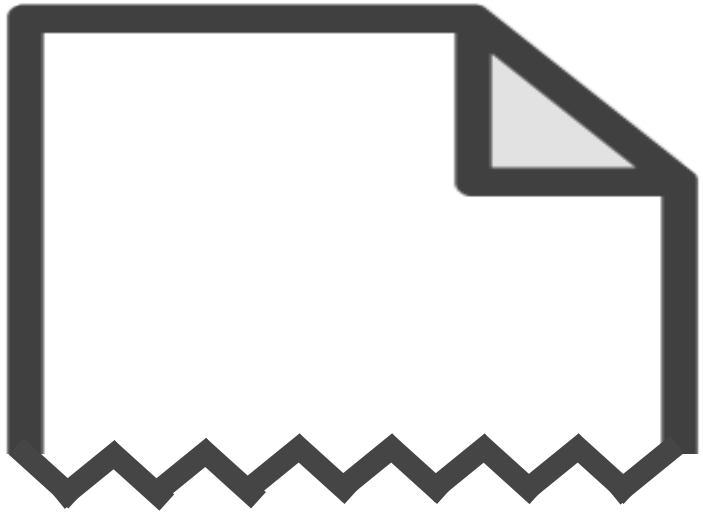
(default = -1 (unlimited))

log.retention.[hours/minutes/ms] (per partition)

(default = 168 hours)

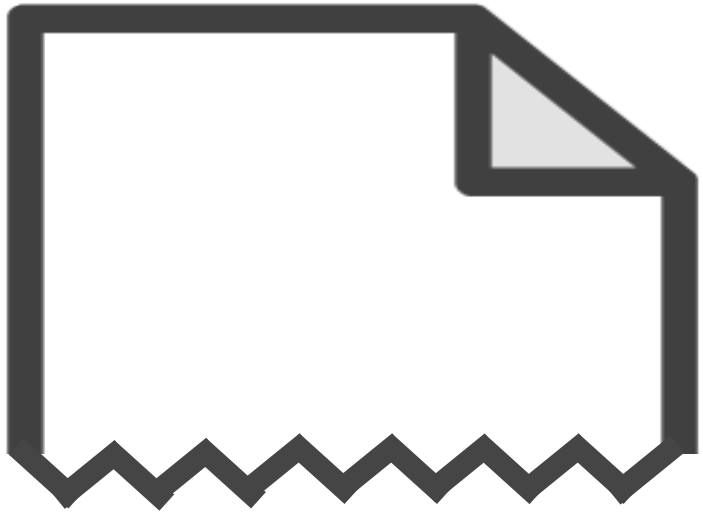


log.segment.bytes



log.segment.bytes

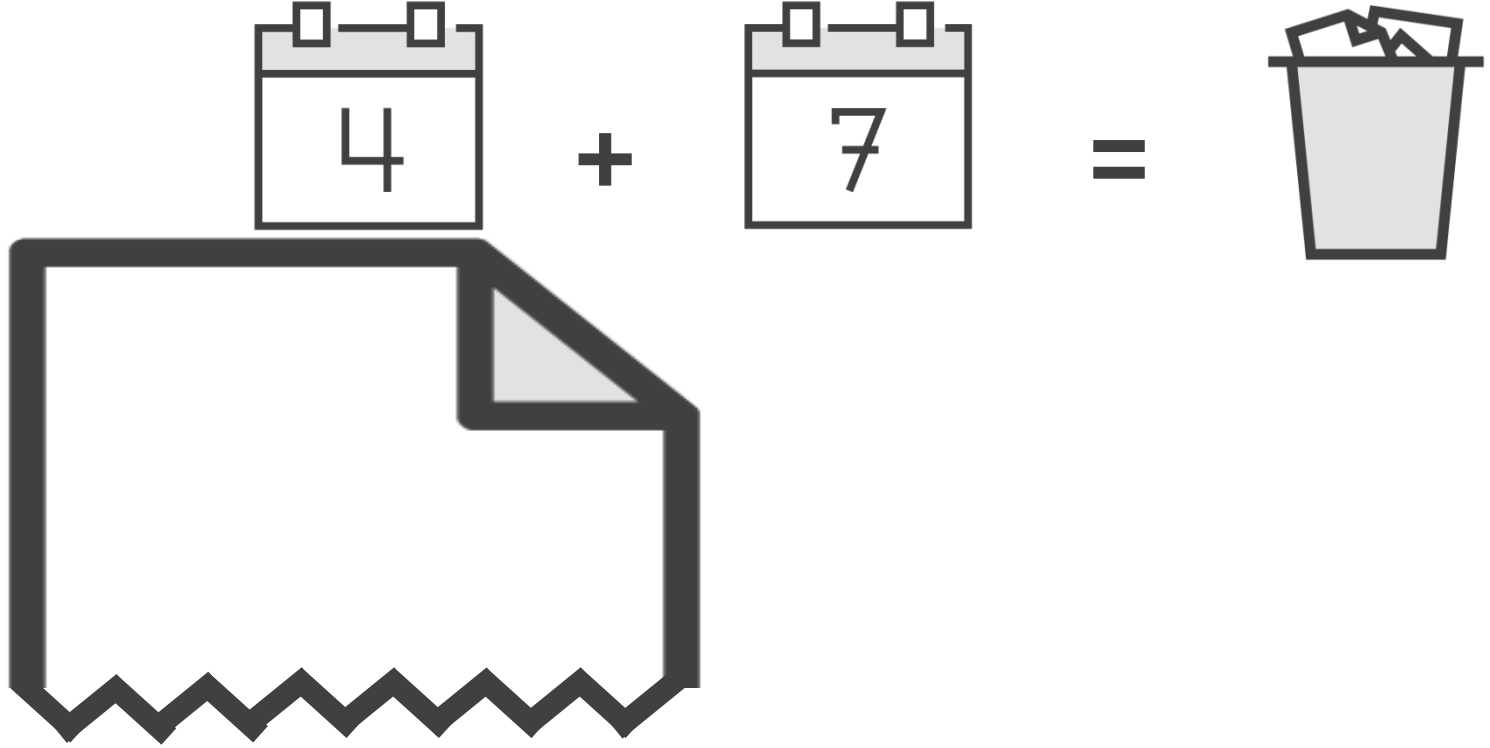
log.segment.ms
(default = null)



log.segment.bytes

log.segment.ms
(default = null)

log.segment.bytes -> 1GB



log.retention.hours -> 7 days
log.segment.bytes -> 1GB

Resources

- How to choose the number of topics/partitions
  - confluent.io/blog/how-choose-number-topics-partitions-kafka-cluster
- Kafka compaction
  - kafka.apache.org/documentation/#compaction
- More on replication
  - confluent.io/blog/hands-free-kafka-replication-a-lesson-in-operational-simplicity

Summary

Broker

- Scale
- Parallelization
- Fault Tolerance
- Data Retention